Improving Training on the Glass-Cockpit CDU Interface

Sharon McLennan (Irving), lionhead@casco.net
James E. Irving, United Airlines 777 Captain, Retired
Peter G. Polson, Institute of Cognitive Science, University of Colorado, Boulder CO 80309-0344
Marilyn Hughes Blackmon, Institute of Cognitive Science, University of Colorado, Boulder, Colorado 80309-0344

ACKNOWLEDGMENTS

This research was supported by Federal Aviation Administration under contract DTFA01-91-C-00028. Earlier reports of this research (Polson, Irving, &Irving, 1995a,b) were written while Peter Polson was a sabbatical visitor at NASA Ames supported by a National Research Council Senior Research Associate Fellowship. The preparation of this report was supported in part by Cooperative Agreement NCC 2-1104 to P. G. Polson from NASA Ames Research Laboratory. The opinions express in this paper are those of the authors and not necessarily the FAA, NASA, or the Federal Government.

We wish to thank the following people for their contributions to our work:

William Traub, United Airlines
John Meyer, United Airlines
Chuck Arthur, United Airlines
Derek Fitch, United Airlines
Greg MacArthur

PUBLICATION HISTORY

This paper is based in part on an FAA Final Report (Polson, Irving, and Irving, 1995b). Brief summaries of the model and the evaluation study were published in Irving, Polson, and Irving (1994) and Polson, Irving, and Irving (1995a, 1995b).

Irving, S., Polson, P., & Irving, J. E. (1994). A GOMS Analysis of the Advanced Automated Cockpit. *CHI '94 Conference proceedings: Human factors in computing systems* (pp. 344-350). New York: ACM.

Polson, P., Irving, S., and Irving, J.E. (1995a). Formal models of human-computer interaction in the advanced cockpit. In R. Jensen & L. A. Rakovan (Eds.), *Proceedings of the Eighth Symposium on Aviation Psychology* (pp. 304-309). Columbus, OH: Ohio State University.

Polson, P., Irving, S., and Irving, J. E. (1995b). Final Report: Applications of formal models of human-computer interaction to training and use of the control and display unit. Federal Aviation Administration, Contract No. DTFA01-91-00028.

CONTENTS

Improving Training on the Glass-Cockpit CDU Interface

Sharon McLennan (Irving), James E. Irving, Peter G. Polson,
and Marilyn Hughes Blackmon

ABSTRACT

We designed a 5-hour CDU training program based on the ACT-R theory of skill acquisition and associated principles for designing intelligent tutors. After training, experimentally trained pilots successfully completed all FAA-mandated CDU tasks in a full-motion simulator test comparable to the FAA checkride. Experimentally trained pilots' performance approximated that of traditionally trained pilots, who had spent 10-50 hours training on sophisticated simulators. Our training design can be applied to teaching flightcrews the full range of CDU tasks, and its time- and cost-efficiency demonstrates the feasibility of teaching substantially more CDU tasks and topics within current airline budgets for CDU training.

1 INTRODUCTION

There is a practical need for improved training on the CDU interface to the FMC. During the 1990s a broad consensus emerged about the necessity of improving flightcrew training for managing the FMC and about the specific topics that need to be added to the curriculum (Air Transport Association, 1997, 1998, 1999; BASI, 1998; FAA Human Factors Team, 1996; Funk, Lyall, and Suroteguh, 1999). Despite the convincing case for expanding the training curriculum to accomplish many additional training objectives, there is only one feasible way to expand the training curriculum for the flightcrew-automation interface. It requires first discovering a way to teach the current

---

[1] lionhead@casco.net

United Airlines 777 Captain, Retired, irving@plv2.innercite.com

Institute of Cognitive Science, University of Colorado, Boulder 80309-0344

curriculum in less training time on less expensive equipment, thereby freeing training time and financial resources to teach more topics. Continuous and intense economic pressure inhibits the development of longer or more expensive training sequences (FAA Human Factors Team, 1996). Syllabuses of transition training programs for "glass cockpit"/FMC aircraft are already overloaded, and current training programs already must devote considerable time on expensive equipment (full-motion or fixed-based simulators).

Our research problem, accordingly, was how to design training programs for the flightcrew-automation interface that are substantially more time- and cost-efficient, and we found a potential solution in the research on designing ACT-R intelligent tutors. These intelligent tutors put into action Anderson’s well-developed ACT-R theory of skill acquisition (Anderson, 1993; Anderson & Lebiere, 1998), and mature ACT-R tutors have repeatedly raised students to a substantially higher level of skilled performance in significantly less time than regular classroom instruction for the same subject. Therefore, we modeled the curriculum and pedagogy for our computer-based training program on the design principles for developing ACT-R intelligent tutors (Anderson, Corbett, Koedinger, & Pelletier, 1995; Corbett, Koedinger, & Anderson, 1997).

Section 2 will explain the design principles and skill acquisition theory we used to design an experimental training program to teach the CDU tasks that are currently required for the FAA checkride. The experimental training program was designed to use inexpensive part-task trainers and to be completed in a small fraction of the time pilots normally spend on learning these same CDU tasks during transition training for "glass cockpit"/FMC aircraft. Section 3 explains the rigorous test used to evaluate the experimental training program: a realistic flight scenario performed in a full-motion simulator. Analysis and discussion of the results occupies Section 4, showing that the performance of experimentally trained pilots compares favorably with the performance of traditionally trained pilots on the same rigorous test. Section 5 draws conclusions and extracts the implications of this successful training experiment in relation to the larger, emerging-consensus goal of expanding the aircrew training curriculum for advanced automation and improving flightcrew understanding of how the automation works.

2 DESIGN OF TRAINING PROGRAM BASED ON ACT-R THEORY OF SKILL ACQUISITION

A major component of the current flightcrew-automation curriculum is training flightcrews to program the FMC through the Control Display Unit (CDU) interface. The specific focus of the work reported here has been to find ways to make the current CDU training component markedly more time- and cost-effective. To design the CDU training program we adhered to the principles used for designing the ACT-R intelligent tutors – eight design principles that have stood the test of time (Corbett, Koedinger, & Anderson, 1997). We intentionally avoided building an actual intelligent tutor, because we believed we could elude the time-consuming process of creating an intelligent tutor and still capture most of the benefits, notably the significant reductions in training time. Our aim was a pragmatic solution that could be implemented without delay in today's aviation training programs, using instructional tools already familiar to and widely used by the aviation community. Section 2.1 describes how we applied the first of the eight design principles, the one that defines the curriculum. Section 2.2 describes how we applied the remaining seven design principles, the principles for designing the pedagogy to deliver the curriculum most effectively.

2.1 Defining the curriculum: Representing competence with a fine-grained model

The aviation community is no stranger to the value of representing aviation competencies with fine-grained cognitive task analyses. The Advanced Qualification Program (AQP) mandates the use of job task analysis and subtask analysis for curriculum development (Longridge, 2000), and aviation trainers increasingly rely on cognitive task analysis for developing aircrew training programs, for designing automated aviation systems, and for managing human resources (see extended reviews by

Seamster, Redding, & Kaempf, 1997, and Schraagen, Chipman, & Shalin, 2000). Cognitive task analysis is superior to traditional behavioral task analysis for tasks and systems that place heavy cognitive demands on aircrews. Such tasks require extensive practice, assimilation of large amounts of knowledge, significant decision making or problem solving, and/or adapting to changing situations under time pressure. A prime example is training pilots to operate the CDU interface to the FMC in an advanced automated aircraft, such as the Boeing 737-300 that was the focus of our experimental training program. Ultimately pilots need to also acquire accurate mental models of the FMC and how the FMC responds to pilot interactions with the CDU interface.

Although cognitive task analysis is widely used for the purpose of designing the content of aviation training programs, our research program is unique in the way it nests the cognitive task analysis within a modern theory of skill acquisition, ACT-R (Anderson, 1993; Anderson & Lebiere, 1998; Anderson & Schunn, 2000). The ACT-R theory of skill knowledge makes a fundamental distinction between procedural knowledge ("how to do it" memory) and declarative knowledge (memory of facts and events). ACT-R, like many other cognitive theories, assumes that procedural knowledge can be represented as a set of all the condition-action rules that are stored in a person's memory when that person has become competent (or expert) in performing a particular skill. With advancing levels of skill, a person acquires a growing repertoire of component physical and mental actions for executing complex tasks, and, for each action, a condition-action rule specifies the condition(s) that must be present to trigger the skilled individual to perform the action.

To represent the competence to be taught in our experimental training we used a detailed NGOMSL model, a variant of the well-known GOMS method (Kieras, 1997; John & Kieras, 1996a, 1996b). Fine-grained NGOMSL analyses have proven an optimal first step towards writing the full-scale ACT-R computer program required to run an ACT-R intelligent tutoring system, as well as for developing computer simulation models of skilled performance used in research to test the ACT-R theory (Anderson & Lebiere, 1998). NGOMSL generates a psychologically valid, very fine-grained analysis, specifying the detailed sequences of physical actions and mental operations necessary to carry out each specific task.

Fortunately, NGOMSL could be quite quickly learned by an aviation trainer or curriculum designer. Indeed, NGOMSL has proved useful for many real-world design situations and has been rapidly and successfully taught to software engineers who are not specialists in either cognitive task analysis or in research on cognitive skills (Kieras, 1997; John & Kieras, 1996a, 1996b). NGOMSL uses natural language and can be learned without prerequisite computer programming skills or graduate training in cognitive psychology.

2.1.1 Overview of NGOMSL

GOMS is an acronym for *Goals*, *Operators*, *Methods* and *Selection rules*, and NGOMSL is an acronym for a *Natural Language* GOMS notation (see Kieras, 1997, for a tutorial on how to use NGOMSL). A *goal* is what the user intends to accomplish, and a *method* is a series of steps that use *operators* to accomplish a goal or subgoal. The NGOMSL analyst performs a top-down, breadth-first decomposition of each goal/subgoal and describes step-by-step methods required to accomplish each goal/subgoal. Typically, a method for accomplishing a high-level goal includes steps that stipulate accomplishment of a subgoal, in addition to steps that use operators. To create a fine-grained NGOMSL model the analyst continues the decomposition process down to lower and lower levels of the hierarchy until the steps in the methods to accomplish the lowest-level goals consist only of operators, not subgoals. In cases where two or more alternative methods are available for accomplishing the same goal/subgoal, *selection rules* govern how to choose among the alternative methods.

At the end of the analysis, all skills that must be learned in the course of the training program should be described in great detail – as step-by-step sequences of simple, observable perceptual-motor actions (for example, pressing a particular key on the CDU) and simple, usually non-observable mental actions (such as a pilot recalling from long term memory the ICAO identifier for the destination before entering the identifier into the CDU). The NGOMSL analyst may choose to use some higher-level, unanalyzed operators for those complex skills that trainees already know how to perform when they enter the training program, such as reading or verifying the correctness of information.

2.1.2 NGOMSL analysis of a pilot's competence using the CDU interface on the FAA checkride

The training program (and the task analysis performed to design the training) targeted the set of tasks that must be passed on the FAA checkride by pilots transitioning to an advanced automated aircraft from an aircraft that is not FMC-equipped. This set of tasks includes the **Preflight FMC** task, which is actually a sequence of six preflight tasks, and eight inflight tasks that require pilots to modify the flight path in response to directives from Air Traffic Control (ATC) – a total of 14 tasks. These 14 tasks compose a tiny fraction of the total number of CDU preflight and inflight tasks that pilots may be required to perform on the CDU interface during line experience. Figure 1 shows the CDU interface used to perform the 14 CDU tasks taught in training.

The completed NGOMSL analysis yielded a hierarchical representation of all the procedural and declarative knowledge that transitioning pilots must acquire during training in order to perform these 14 preflight and inflight tasks. To understand expert performance of these 14 tasks we mapped the relevant skills and knowledge of a subject matter expert who was one of the co-authors of the study (J.I.). At the time of this study, J.I. was a Boeing 737-300 Captain, and he had also been an instructor and first officer on the Boeing 757/767. To do the research described here, all of the authors became adept in NGOMSL, including our subject matter expert. This substantiates our claim (see above) that the NGOMSL analysis and training design principles used for this research project are both practicable for use by aviation course designers and broadly applicable for developing flightcrew training programs.

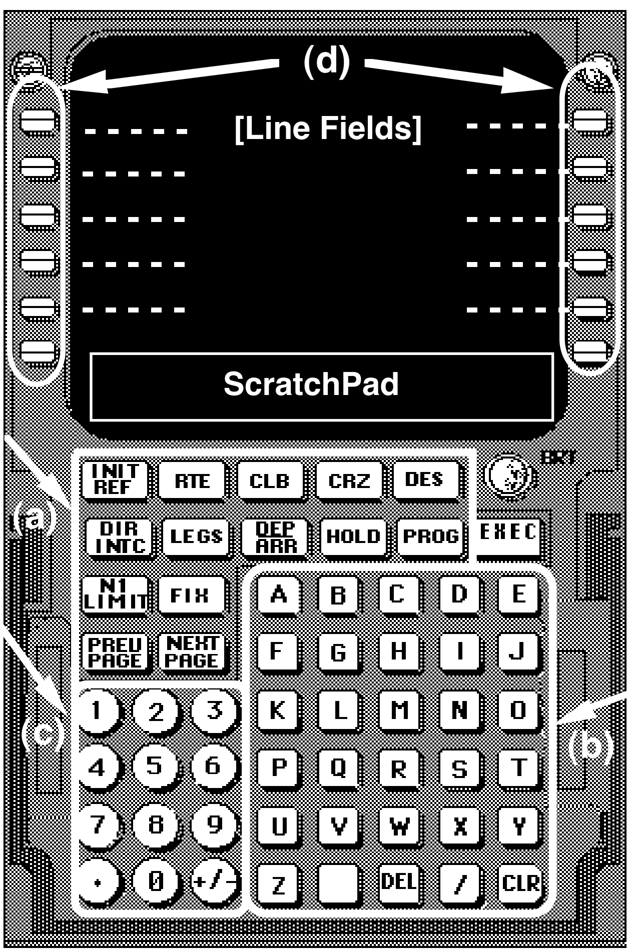

Figure 1. Control Display Unit (CDU)

Due to space constraints we will restrict coverage here to a few components of the full NGOMSL model that illustrate the eight principles for designing an intelligent tutor (for a more complete NGOMSL model of the CDU tasks, see Polson, Irving, & Irving, 1995, Appendix A).

2.1.3 Simple example of NGOMSL methods: INSTALL HOLD

Figure 2 shows a method for accomplishing the goal INSTALL HOLD, one of the inflight **Modify Route** tasks. INSTALL HOLD illustrates a simple but typical CDU method, one that contains a mixture of steps that set subgoals and steps that evoke simple or unanalyzed operators. The methods for accomplishing INSTALL HOLD and other CDU tasks involve entering or verifying required information on a specific page displayed on the CDU. Step one, for example, requires the pilot to retrieve and retain the waypoint that ATC specified for the holding

Method to Accomplish goal of INSTALL HOLD
1) Retrieve the Hold-at waypoint and retain.
2) Accomplish the goal of **ACCESS** [HOLD area].
3) Accomplish goal of **DESIGNATE** [Hold-at waypoint].
4) Accomplish goal of **INSERT** [Hold-at waypoint].
5) Accomplish goal of INSTALL HOLD PARAMS.
6) Verify.
7) Execute.
8) Report goal accomplished.

Figure 2. Method to accomplish goal of INSTALL HOLD

fix (for example, retrieved from the Air Traffic Control directive and retained in working memory). Steps six and seven specify unanalyzed operators – verify and execute – that the pilot knew how to do from previous experience before entering training.

In contrast to the three steps evoking simple/unanalyzed operators, four steps each set a subgoal, beginning with the words "Accomplish goal of…." Three of these four steps (steps two, three, and four) each calls for accomplishing a core intermediate-level goal – **ACCESS, DESIGNATE,** or **INSERT** – that appears repeatedly across almost all of the methods used on the CDU. The pilot must perform **ACCESS, DESIGNATE,** and **INSERT** so often to accomplish so many different tasks on the CDU that the methods for accomplishing these three core goals become highly practiced and are an important source of transfer across tasks (see below, Section 2.1.4). The last step, step eight, reports the goal accomplished, returning control to the higher-level goal, the one that originally called for accomplishing the goal INSTALL HOLD.

2.1.4 Well-organized hierarchical goal structure

Due to the top-down, breadth-first approach, the work of constructing a NGOMSL model starts with the top-level goal, moves down to the second-layer subgoals set by the top-level goal, then moves down to the third layer subgoals for accomplishing second-layer subgoals, and so forth. The process of constructing a NGOMSL model results in a well-organized hierarchical goal structure topped by a single top-level goal. NGOMSL thus derives the organized structure of the skill, not just the individual physical actions and mental operations required for performing the skill. Furthermore, the training sequence designed from the NGOMSL model reflects and transmits the same hierarchical goal structure.

The top-level goal for flying the aircraft, shown in Figure 3, uses a task acquisition loop and the methods needed to manage prioritization and execution of multiple tasks in the cockpit. Figure 4 displays the top-level goal structure for flying the aircraft and nests within it the CDU tasks to be taught during training. The **Preflight FMC** and **Modify Route** goals – the focus of the NGOMSL analysis – are displayed in Figure 4 as yet-unpacked boxes. The **Preflight FMC** goal is one of many preflight goals that the aircrew accomplishes before taking off and unpacks into six CDU tasks. The

Method to Accomplish Goal of Navigate-Origin-to-Destination
Do until no more tasks
Begin loop
    Accomplish goal of Acquire-New-Task
    Accomplish Goal of Assess-Current-Situation
    Accomplish Goal of Perform-Highest-Priority-Task
End loop
Report Goal Accomplished

Figure 3. Method to accomplish top-level goal of flying the aircraft

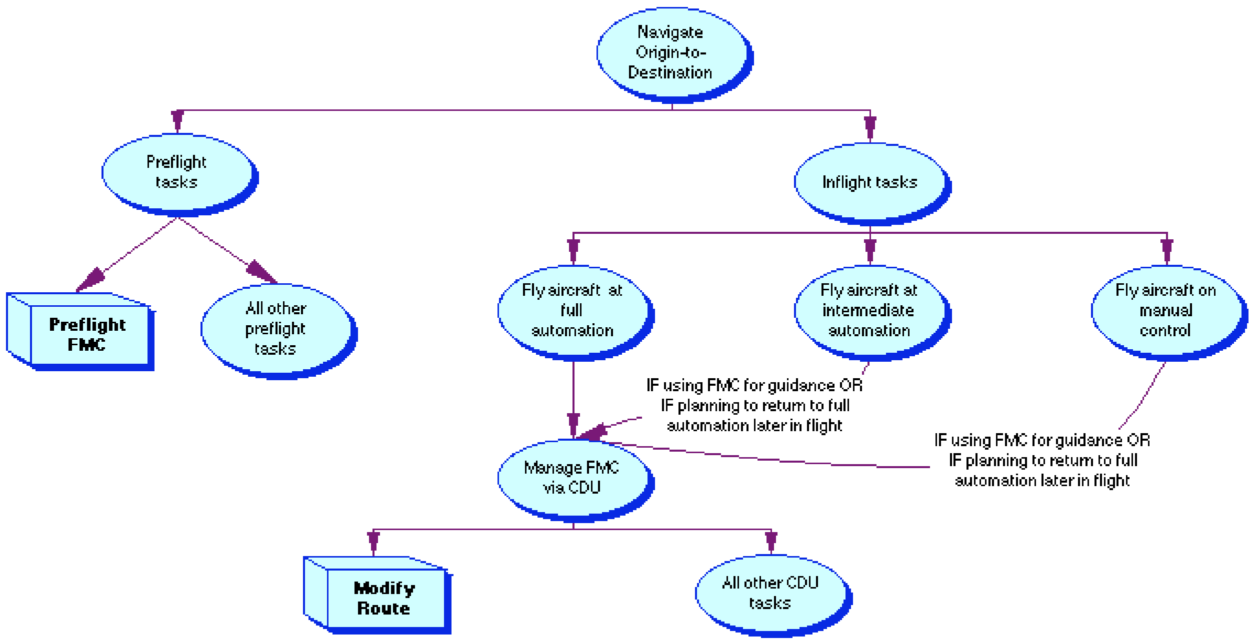

Figure 4. CDU tasks nested within top-level goal structure for flying the aircraft

Modify Route goal is one of many goals to be accomplished while airborne, and **Modify Route** unpacks into a large set of inflight CDU tasks that aircrews can be required to perform during flight. The eight inflight tasks that are tested on the FAA checkride compose only a tiny fraction of all inflight CDU tasks.

Expert pilots perform inflight CDU tasks in the context of a multitasking environment driven by the hierarchical goal structure. For the top-level goal Navigate-Origin-to-Destination, the subgoal Perform-Highest-Priority-Task can interrupt performance of a lower-priority task. The tasks acquired include receiving directives from Air Traffic Control (ATC), which have high priority in any non-emergency inflight situation. The pilot is responsible for adjusting the response to the ATC directive to fit the flight circumstances. Pilots accomplishing the **Modify Route** goal in actual inflight situations must first decide which level of automation to use to fly the aircraft – full, intermediate, or manual. Pilots accomplish the **Modify Route** goal in order to comply with the ATC directive while using the automation to fly the aircraft.

All inflight CDU tasks must be learned in the context of realistic flight scenarios so that the pilot gains a clear understanding of the complex possible interactions among the pilot, autopilot, and FMC during flight. For example, the pilot can be manually flying the aircraft and still want guidance from the Flight Director. If a pilot is flying the aircraft manually but using the Flight Director, the pilot is not using the automation fully but still needs to program the FMC with the route modifications. Even if the pilot is using none of the automation, the pilot must modify the route to keep the FMC current if s/he expects to use the automation later in the flight (although such a task can often be deferred to a period of lighter workload). In case of emergency, however, the pilot must not take time to program the FMC and may need to turn off the automation. With sufficient practice the pilot can quickly make the appropriate decisions in each possible flight situation.

When the full-scale NGOMSL analysis unpacked the subgoals of the **Preflight FMC** and **Modify Route** goals, it revealed two potential sources of savings in training time due to transfer of training. Subsection 2.1.4.1 describes the first source: three common subgoals that are called repeatedly by almost all higher-level goals. Subsection 2.1.4.2 explains the second source of transfer: the subordination of highly similar goals under a higher-level goal to expose both their similarities and distinguishing features.

2.1.4.1 Identifying core subgoals for accomplishing diverse goals

Constructing our NGOMSL model enabled us to discover something important about the current CDU interface: a set of three common intermediate-level subgoals – **ACCESS**, **DESIGNATE** and **INSERT** – that are repeatedly called by the various methods for accomplishing almost every higher-level CDU goal. These three common subgoals (and the methods for accomplishing them) create the potential to save learning time by increasing transfer – assuming the training syllabus facilitates transfer by drawing attention to these three common subgoals. This discovery has crucial implications for training. Instead of teaching each CDU task as a rote sequence of keystrokes and their related mental operations, pilots are taught to accomplish subgoals that each substitutes for a block of keystrokes and linked mental operations.

We designed our training sequence to ensure that pilots would recognize instantly when to accomplish each of these three core subgoals, and master the various methods for accomplishing them. Accomplishing the **ACCESS** goal involves two sets of cognitive functions: (1) identify the page associated with the current task, and then (2) carry out the sequence of operations necessary to get that page (screen) displayed on the CDU. The **DESIGNATE** goal is invoked any time the pilot must enter parameters into the CDU interface. The method to accomplish this goal models information flows, because information to be designated (i.e., information required to complete the task) is retrieved from various sources – including dispatch paperwork, the airport terminal information service (ATIS), Air Traffic Control (ATC) directives, the CDU itself, or long term memory. Accomplishing the **INSERT** goal involves inserting the information that was "designated" (and thus currently in the CDU ScratchPad) into the proper line (by determining which is the proper line and then pressing the corresponding line select key (LSK 1 - 6 left, or LSK 1 - 6 right). At this point, the system runs a check on the inserted data, rejecting it with a message if its value is out of range, improperly formatted or not found in the database.

The instructional materials emphasized these three common goals and their associated methods throughout the training sequence. The first two training modules introduced the intermediate-level subgoals **ACCESS**, **DESIGNATE** and **INSERT**. After pilots learned the methods for accomplishing these three subgoals, they quickly moved on to accomplishing these subgoals in service of accomplishing the higher-level **Preflight FMC** goal. After that pilots advanced to accomplishing these same three subgoals in the context of inflight **Modify Route** tasks. Emphasizing these three core subgoals during training elicited transfer of training across all the diverse tasks that call these three subgoals.

Figure 5 shows the goal structure for the higher-level **Preflight FMC** goal, which calls for accomplishing six main subgoals: CHECK IDENT, POSITION INITIALIZATION, INSTALL ROUTE, PERFORMANCE INITIALIZATION, REVISE V-NAV CLIMB, and TAKE-OFF REF. The step-by-step methods for accomplishing these six subgoals, in turn, call for accomplishing the three intermediate-level subgoals. For example, INSTALL ROUTE begins by calling the intermediate-level subgoal **ACCESS** (to **ACCESS** the Route page) and then selects one of two subgoals, either INSTALL COMPANY ROUTE or INSTALL MANUAL ROUTE. Regardless of which of the two subgoals is chosen, the method of accomplishing the selected goal will call the other two intermediate-level subgoals **DESIGNATE** and **INSERT**. Following that, the main INSTALL ROUTE goal recovers control and calls the two intermediate-level subgoals – **DESIGNATE** and **INSERT** – to **INSERT** the Departure and Arrival routing and then **DESIGNATE** and **INSERT** the departure runway.

The intermediate-level goals **ACCESS**, **DESIGNATE** and **INSERT** can, in turn, call lower-level goals. Experienced pilots transitioning from non-CDU aircraft such as the B-727 or DC-10 already know how to accomplish many lower-level goals, such as the complex actions VERIFY, ACTIVATE, EXECUTE, CHECK, DETERMINE, COMPARE, SELECT, RECALL (from long term memory), RETRIEVE, and RECEIVE ATC DIRECTIVE. Pilots did not need to learn methods for accomplishing these goals during training, so the NGOMSL model treats them as unanalyzed higher-level operators. The training program taught pilots the methods for accomplishing lower-level goals

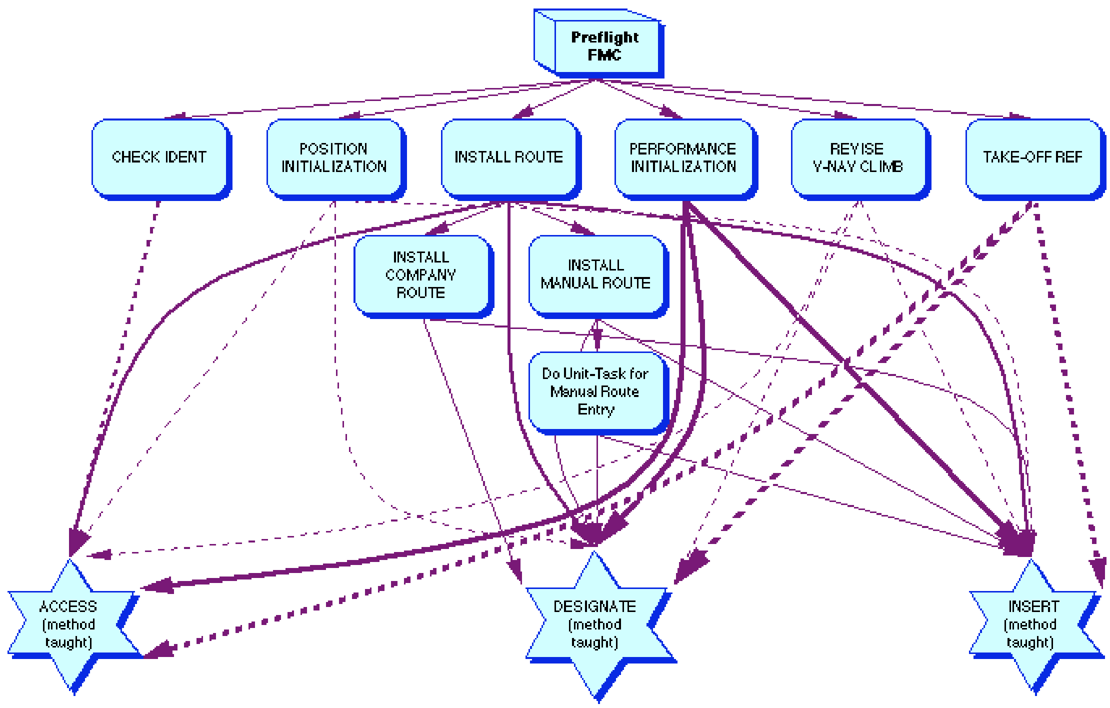

Figure 5. Preflight FMC goal unpacked into subgoals down to low-level goals and basic operations

first encountered during training (e.g., COPY DATA and ENTER DATA) and simple primitive operators (e.g., KEYPRESS Specific Function Key or KEYPRESS Specific Line Select Key). Therefore, the net effect of accomplishing any of these three intermediate-level goals is to apply a well-practiced chunk/sequence of actions – similar to a subroutine in computer programming – in service of accomplishing a higher-level goal. The ultimate effect is to simplify the methods for accomplishing CDU tasks and maximize transfer among all the various CDU tasks.

2.1.4.2 Grouping related tasks under a higher-level goal

Another important insight that emerged from the NGOSML analysis was that it is inefficient to teach pilots the inflight CDU tasks as isolated tasks (the way these tasks have been presented in conventional CDU training sequences). Instead, an optimal training course would teach the eight inflight tasks as eight distinct methods for accomplishing a single high-level goal: **Modify Route**. Our CDU training taught pilots explicit selection rules for distinguishing which particular method to use to modify the route under each specific set of circumstances. The goal hierarchy for the **Modify Route** tasks is shown in Figure 6.

With this reorganized structure, **Modify Route** then encompasses all eight of the following tasks: (1) DIRECT-TO (on route), (2) DIRECT-TO (off route), (3) INTERCEPT LEG-TO (on route), (4) INTERCEPT LEG-TO (off route), (5) INSTALL AIRWAY, (6) INSTALL HOLD, (7) EXIT HOLD, and (8) INSTALL APPROACH. The methods for accomplishing these eight tasks are very similar, and grouping them together draws attention to the similarities, fostering transfer of training and thereby reducing training time.

Just as important, grouping these eight similar tasks into a single higher-level goal highlights the distinctive components of each method, not just the similarities. The most crucial distinctive features are the eight selection rules that determine which of the eight methods to apply in any given situation, corresponding to eight distinct Air Traffic Control (ATC) directives. The vast majority of

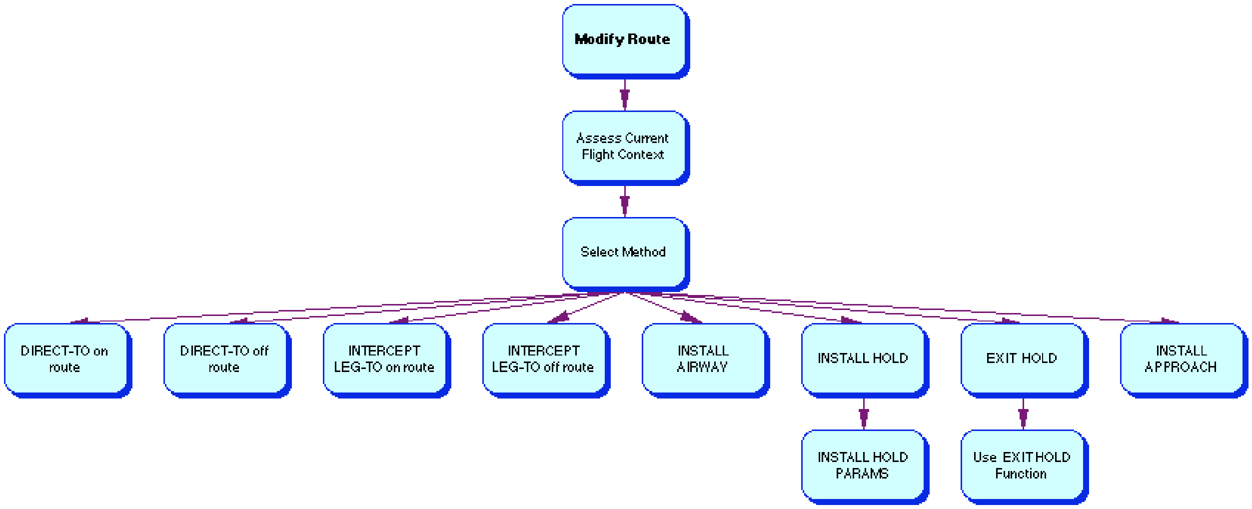

Figure 6. Goal hierarchy of subgoals for eight Modify Route CDU tasks performed during flight.

tasks in flight are driven by communication from air traffic control (ATC) in the form of directives to change the originally cleared routing. Teaching pilots the eight tasks as eight methods for accomplishing a single goal – **Modify Route** – results in pilots rapidly mastering the subtle distinctions among the ATC directives and reliably responding with the correct method in response to each distinct ATC directive.

2.2 Design the most effective pedagogy to deliver the curriculum

The first ACT-R principle for designing an intelligent tutor builds a fine-grained model of competence that defines the curriculum, and the remaining seven principles determine the most effective pedagogy for delivering that curriculum. Section 2.1 (above) described the NGOMSL model that defined the curriculum for our training program, and Sections 2.2.1 to 2.2.7 (below) will describe these seven principles and how they were implemented in our training design.

2.2.1 Promote abstract understanding of the knowledge needed to perform skills

The first of the seven pedagogical principles for designing an intelligent tutor is to foster transfer of skills to novel problems by promoting students' acquisition of procedural and declarative knowledge that is sufficiently general for solving a broad class of problems. A consistent finding in the research literature on skill acquisition is that learners who create a hierarchical goal structure and better understand the underlying abstract principles that structure and integrate the domain are able to solve a wider range of near- and far-transfer problems (Chi et al., 1989, 1994; Pennington, Nicolich, & Rahm, 1995; VanLehn, Jones, & Chi, 1992). Students tend to develop overly specific knowledge that transfers poorly if they (1) learn a rote sequence of procedures, (2) study a single example problem, and/or (3) limit problem-solving practice to a set of highly similar problems.

Our training program used three strategies to comply with this principle. First, for each of the 14 CDU tasks, the realistic flight scenarios posed a diverse array of practice problems, encouraging pilots to encode procedures general enough to cope with almost any instance of that CDU task that they might ever encounter. Second, instead of learning rote procedures applicable to just one CDU task, pilots learned three subgoals applicable to any CDU task – **ACCESS, DESIGNATE,** and **INSERT** – and a very flexible family of methods for accomplishing each of these subgoals. Third, we connected the newly acquired set of CDU skills to pilots' previously acquired higher-level knowledge by nesting the goal structure for the CDU skills within the larger context of the multi-tasking cockpit environment (see above, Figures 3 and 4). For example, the eight selection rules for **Modify Route** tasks integrate pilots' CDU knowledge with their knowledge of ATC directives. In the future, training programs for the flightcrew-automation interface will need to take this strategy even further, integrating a sophisticated mental model of the FMC into pilots' knowledge for flying the aircraft.

2.2.2 Clearly communicate the goal structure of the task

The second of the seven pedagogical principles for designing an intelligent tutor is to focus attention on the goal structure of the task. When a person successfully solves a problem, s/he first creates a problem space and sets subgoals that make it possible to move from the initial state to the goal state. Successful problem solving depends on how well the problem solver understands the goal structure, and a skilled problem solver quickly accomplishes the series of subgoals and reaches the end state (Newell, 1980; Anderson, 1993). An intelligent tutor can communicate the goal structure by explicitly representing it (on the computer screen, for example) and/or by communicating it through help messages.

Our training design communicated the goal structure both ways. First, from the outset of training the computer-based training (CBT) taught and reinforced the three core subgoals. Instead of learning each CDU task as a sequence of up to 18 keystrokes, pilots learned to perform a set of subgoals that are common to all CDU tasks and a family of closely related methods for accomplishing each subgoal. In Session I the CBT repeatedly called learners' attention to the three core subgoals, emphasizing formulation of correct goals in response to ATC directives before pressing any keys. The CBT tutorial prompted learners, if necessary, to **ACCESS** the appropriate area (the program waited for the correct response), then **DESIGNATE** the appropriate route element or information (program again waited for the correct response), and then **INSERT** that information on the correct line (program waited for correct response). Whenever the CBT asked learners to actually carry out one of these goals, the CBT consistently displayed the name of the goal in red font. The training also taught the selection rule for the eight methods nested under the high-level **Modify Route** goal, analogous to the six methods for accomplishing the subgoals set by the **Preflight FMC** goal.

Second, our training design communicated the goal structure via help messages. When the learner needed a hint the CBT provided help messages that would initially just remind the learner of the current goal, usually **ACCESS**, **DESIGNATE** or **INSERT**. If that reminder proved insufficient, the CBT offered hints about how to accomplish the goal.

2.2.3 Have students learn primarily by doing – by solving problems

The third of the seven pedagogical principles for intelligent tutor design is for students to learn by doing. Solving problems results in the acquisition of condition-action rules that get stronger with repeated practice (Anderson & Lebiere, 1998, Chapters 2 & 4). Pilots enrolled in our training program learned almost exclusively by using the CDU to solve problems framed in realistic flight scenarios, spending most of Session I and all of Session II solving problems. Starting early in Session I and continuing through the end of Session II, they had to figure out how to accomplish the **Preflight FMC** goal for several different airline routes. After completing the first **Preflight FMC** they began doing the less routine, more challenging assignments of accomplishing the **Modify Route** goal in response to inflight ATC directives for a wide variety of realistic flight scenarios. The set of problems solved are representative of those on which transitioning pilots must perform well during the FAA check ride (the particular flight scenarios solved during training are available in Polson, Irving, and Irving, 1995, Appendix C).

Using realistic flight scenarios is a crucial guideline for designing part-task trainers for use in aviation training (Eurocontrol, 2000). Moreover, our part-task training equipment faithfully represented the CDU that pilots would use in the cockpit or in the full-motion simulator. The CDU system displayed on the computer screen responded in the same manner as the actual device, enabling pilots in training to carry all action sequences pertinent to the task with appropriate feedback from the display. For example, when carrying out the actions for a DIRECT-TO, there were changes in value and position for the information relating to the appropriate point on the route, the course and distance to that point, and all four of the remaining points shown on that "page" of the CDU display.

Airline training programs generally use full-motion or fixed-base simulators to train pilots on the CDU, but ACT-R theory claims that the condition-action rules acquired by the end of training will be virtually the same regardless of whether pilots learn by solving problems in a full-motion simulator, a fixed-base simulator, or on part-task trainers. Thus, part-task trainers should result in acquisition of the same skill – the same set of condition-action rules – while realizing considerable savings in equipment expenditures. During Session II, experimentally trained pilots jumped down the route scenarios as a series of snapshots, so they were able to spend all of their time solving problems. In contrast, in full-motion simulators it is necessary to fly a route in "real time," increasing the time taken to go from one critical learning event to another. The CBT Authorware Professional™ and SuperCard™ software programs used in our training program collapsed these irrelevant "real time" periods and simply informed the pilot that the plane s/he was flying was now at a waypoint farther down the route. Then the pilot immediately began working on the next problem. Fixed-base simulators can be set to fly twice or four times as fast, compressing the time between waypoints but still devoting a far smaller percentage of the training time to solving problems than our experimental training did.

1.1.4 Gradually increase the grain size of the instruction

The fourth of the seven pedagogical principles for intelligent tutor design is to adjust the grain size of instruction with learning. With practice students can advance from low-level goals to accomplishing higher-level goals without being told the individual steps for accomplishing these goals. In accord with this principle there was a clear progression in the grain size of the tasks experimentally trained pilots performed. Training started with coverage of the function and display areas (Scratch Pad, line select keys, function keys) basic operations (for entering, copying, clearing, and deleting data), and the most common methods for accomplishing the three intermediate-level goals, **ACCESS, DESIGNATE**, and **INSERT**. Pilots then assembled the component operations to solve **Preflight FMC** problems, routine tasks analogous to form-filling tasks in an office environment. Following this, pilots were presented with several **Modify Route** problems, learning selection rules to distinguish which particular method to use to respond to the particular ATC directive. Thus, the grain size of instruction in our training program increased quite dramatically as pilots moved through the instructional sequence.

1.1.5 Progress to real-world performance, reducing tutor assistance

The fifth of the seven pedagogical principles for intelligent tutor design is to gradually reduce assistance from the tutor so that learners progress to real-world performance with real-world feedback. Accordingly, the training facilitated successive approximations to the target skill, starting with scaffolding and moving quickly toward real-world performance conditions.

The CBT for Session I, programmed with Authorware Professional™, presented a single-path series of tutorials and precluded free exploration of the CDU, requiring the user to stay with each individual exercise. The training program in Session I was designed to be as free of frustration as possible. No experimentally trained pilot had to sit and puzzle over how to make a response for more than a half minute. Immediate error feedback kept each pilot from wasting time by floundering in error or uncertainty. Nevertheless, as the pilots encountered more CDU tasks, they gradually found ways to adapt the **ACCESS, DESIGNATE**, and **INSERT** methods to fit a new type of **Modify Route** task, resulting in acquiring families of methods to accomplish **ACCESS, DESIGNATE**, and **INSERT** goals. As pilots responded to a more diverse repertoire of ATC directives for inflight tasks, they also gained flexibility and learned to respond appropriately to a wide variation of real-world situations.

Session II rapidly faded out the scaffolding. The SuperCard™ program used to drive Session II allowed pilots to carry out tasks in a much less rigid fashion than the single-path Authorware

Professional™ CBT program used for Session I. The SuperCard™ program allowed pilots to select which particular flight route to do next, **ACCESS** different areas, go back and forth among LEGS pages, and/or navigate around to different areas of the CDU at any time. There was no tutorial of any kind in Session II, but the program required each experimentally trained pilot to complete five or six realistic flight scenarios that were carefully crafted to provide critical training events (for an example see Appendix). An experienced line pilot (one of the co-authors, J.I.) selected these flight scenarios from a large number of actual printouts of flight plans and ATC directives. Experimentally trained pilots had to carry out all the required items with little or no assistance from the tutor.

1.1.6 Provide immediate error feedback to enhance learning and avoid floundering

Providing immediate error feedback is the sixth of the seven pedagogical principles for designing an intelligent tutor. The phrase "immediate error feedback" requires further explanation, because Anderson and his colleague recommend administering feedback incrementally in three or four stages. The first hint should be just a reminder of the goal. If the first-stage help message is not enough, the second help message should describe the relevant features of the current state in the problem space and the end goal. A third hint, if needed, can provide the rule for moving from the current state in the problem space towards the end goal. Only as a last resort should the help message describe a concrete action to take in the situation. Providing feedback in incremental stages also makes allowances for important aptitude treatment interactions. Students with strong background knowledge tend to benefit most from the second level of help message, while students with weak background knowledge benefit most from third-level help messages. In addition, limiting the feedback to signaling the presence of the error (without commenting on it, diagnosing it, or providing the correct solution) gives the student a sense of control.

Anderson (1993; Anderson et al., 1995) reported an experiment that compared student performance on the LISP computer-programming tutor under four different tutoring modalities. Students in the immediate-feedback modality (see further explanation of this modality below) completed the tasks in the least amount of time. Students in the no-feedback modality took about three times as long and students in the feedback-on-demand modality took about twice as long. The error-flagging modality (students had the freedom to ignore the error feedback or request the tutor's error message) took less than twice as long but substantially more than the immediate-feedback modality. Despite these marked differences in time expended solving the same set of problems, the performance of all four groups was virtually the same after the groups had finished solving all the problems. The underlying reason is that solving the problems resulted in acquisition of the same set of condition-action rules, which, in turn, resulted in equivalent performance.

There are many subtleties to designing error feedback, and Anderson and colleagues continue to research and fine tune error feedback to avoid interference with crucial aspects of learning and support deeper understanding (for a review, see Corbett, Koedinger, and Anderson, 1997). For example, too much feedback or feedback that provides the answers makes students dependent on feedback, resulting in a serious drop in performance level when feedback is removed. Optimal feedback supports development of solution-generating skills, error recovery skills, meta-cognitive and self-monitoring skills, and a deeper understanding that increases transfer to difficult, novel problems (Chi et al., 1989).

Since a primary goal of our training experiment was to sharply reduce learning time, we decided to provide immediate error feedback during Session I. In our training program we were careful not to interrupt pilots and their current working memory state to point out minor errors. The Authorware Professional™ part of the CBT (used during Session I) allows some branching with good feedback for errors. The program waits for the correct response and gives feedback for incorrect responses.

The Authorware software allows the programmer to specify the order in which error messages will appear. Thus the CBT presented different kinds of feedback depending on whether it was the first, second, or third incorrect response to a requested action, or a time-out. When the pilot in training needed a hint the CBT would first remind the pilot of the current goal, usually **ACCESS**,

DESIGNATE or **INSERT**. If more assistance proved necessary, the CBT would offer hints about how to accomplish the goal. The CBT would resort to providing explicit directives only in the case where more than two incorrect responses were given to a single exercise.

To promote development of solution-generation, meta-cognitive, self-monitoring, and error recovery skills, the SuperCard™ program for Session II gave pilots considerably more freedom in solving the problems than the CBT used in Session I, greatly reducing error feedback and thereby advancing them towards independently solving real-world problems. The instructor intervened during Session II only if the learner requested help or was seriously floundering.

1.1.7 Minimize working memory load

The last of the seven pedagogical principles for intelligent tutor design is to minimize working memory load during learning, because a high working memory load can interfere with problem solving. Learning complex problem-solving skills can place excessive loads on working memory, but each CDU task can be performed in under a minute and generally does not impose high demands on working memory.[1] Therefore, minimizing working memory load was not crucial for designing the experimental training. Nevertheless, the training design did reduce working memory load by structuring training around the common intermediate-level goals **ACCESS, DESIGNATE,** and **INSERT**. Instead of learning each CDU task as a long sequence of separate keystrokes, pilots acquired organized sequences of steps. These larger, more meaningful units eased memory load during learning.

3 EXPERIMENT TO EVALUATE THE TRAINING DESIGN

The previous section (Section 2) explained the underlying theory and design principles of our training, and this section describes the training experiment designed to evaluate our training program. Our motivation for formally evaluating this training program was three-fold: (1) to test our NGOMSL model of the CDU tasks in order to refine the model for the next iterations of the training, (2) to validate predictions that the training design markedly reduced training time, and (3) to test the hypothesis that the skills represented in the model can be acquired efficiently through cost-effective part-task training – that is, out of context of other cockpit automation, devices, and displays – and confirm whether skills learned in this manner do actually transfer to a full-motion simulator.

3.1 Participants in the experiment

The group of pilots who received the five-hour experimental training program were experienced Boeing 737-200 line pilots from a major airline recruited from flyers in company mailboxes (**experimentally trained pilots**, n=19; one member of the group flew the Boeing 727 instead of the 737-200). None of these pilots had ever had any experience in a "glass cockpit" equipped with an FMC.

A second group of pilots (**traditionally trained pilots**, n = 19) worked for the same airline as the experimentally trained group, and the two groups were closely comparable. The traditionally trained pilots had just successfully completed the transition training for the 737-300, a "glass cockpit" aircraft with a FMC, but had not yet had line experience on this or any other "glass cockpit"/FMC aircraft. We were able to run the traditionally trained pilots within the 24-hour window between their FAA checkride and the line oriented flight training session ("the LOFT") they were required to perform in the full-motion simulator before being released to the line as a 737-300

[1] The INSTALL HOLD PARAMS task is an exception, because the pilot must convert the radial, given in the ATC directive, to its reciprocal (see Section 4.1.2). Some pilots performed the conversion mentally, while others used paper and pencil to aid working memory.

pilot. As estimated by their flight trainers, traditionally trained pilots probably spent anywhere from 10-50 hours on the CDU component of the transition training. This represents two to ten times as long as our five-hour experimental training program.

The third group comprised 737-300 commercial line pilots (**experts**, n = 10), who had been flying this "glass cockpit"/FMC aircraft for the same major airline for more than one year. Like the traditionally trained pilots, the experts had all completed the traditional transition training for a "glass cockpit"/FMC aircraft but at this point had spent many hours performing both preflight and inflight FMC-programming tasks on the CDU during their year or more of line experience.

3.2 Target CDU Tasks

The target tasks were those CDU tasks tested on the FAA mandated check ride – all lateral navigation (L-NAV) tasks only and no vertical navigation (V-NAV) tasks. The FAA-mandated tasks include six preflight tasks that together compose the **Preflight FMC**: (1) CHECK IDENT, (2) POSITION INITIALIZATION, (3) INSTALL ROUTE, (4) PERFORMANCE INITIALIZATION, (5) TAKE-OFF REF, and (6) REVISE V-NAV CLIMB. The mandated tasks also include eight inflight tasks: (1) DIRECT-TO on route, (2) DIRECT-TO off route, (3) INTERCEPT LEG-TO on route, (4) INTERCEPT LEG-TO off route, (5) INSTALL AIRWAY, (6) INSTALL HOLD, (7) EXIT HOLD, and (8) INSTALL APPROACH. This makes a total of 14 separate CDU tasks to be targeted during the transfer test.

3.3 Transfer test procedures and test equipment

All three groups of pilots completed the transfer test of performance on the 14 CDU tasks mandated for the FAA checkride. All pilots were tested individually in a 737-300 full-motion simulator utilizing a realistic line oriented flight training (LOFT) scenario – a 30 minute simulated flight on a company route from Denver to Colorado Springs. The transfer test performance of each individual was videotaped. The transfer test required that the pilot start by carrying out the six **Preflight FMC** tasks. (The pre-flight data was supplied as the pilot requested it rather than provided in simulated dispatch paperwork.) While airborne on the simulated flight the pilot had to respond to ATC directives to modify the route. The test is shown in Figure 7.

Each pilot performed the series of FMC programming tasks as pilot-not-flying (PNF), while an experienced instructor acted as pilot-flying (PF). The PNF in most airline environments is the one who does the FMC work, i.e. the pilot who is actually interacting with the CDU to carry out the FMC tasks. Pilots being tested responded to the clearances by copying them in their own style (on scratch paper) and then reading them back in the normal manner of communications with ATC. Upon completion of the read-back, the timer was started for that task. When the keystrokes representing correct completion of the task were observed, the timer was stopped. If necessary to enable the pilot to complete the task, the instructor-experimenter provided a hint and performances needing no hints were scored as superior to performances where hints were supplied. (A complete description of the transfer test environment, procedures, and scenarios used, is provided in Polson, Irving, & Irving, 1995, Appendix E.)

3.4 Equipment and software for training program

We minimized costs of training equipment by designing the two training sessions for the experimentally trained pilots to use software that runs on inexpensive, readily available personal computers – a fast Macintosh (IIfx or Quadra) with 20 megabytes of RAM and at least a 16” color monitor (256 colors). The Macintosh screen display offered a primitive desktop simulator featuring a realistic simulation of the CDU and a low-fidelity simulation of the horizontal situation indicator (HSI) but no other flight automation or displays found in the actual 737-300 cockpit. We developed a single-path Authorware Professional™ CBT program for Session I and a SuperCard™ program for

"*[Company aircraft]* 123, cleared for takeoff on runway 35L"
Climbing at approximately 1000 feet after takeoff: "*[Company aircraft]* 123, turn right heading ° direct Kiowa, Victor 83, flight plan route" ***[DIRECT-TO IOC (off route); INSTALL AIRWAY V83]****. "IOC" is the identifier for Kiowa. This involved going direct-to a point not on the originally programmed route and then adding an airway to the route.*
A short time later: "[*Company aircraft*] 123, due to traffic in the Colorado Springs area, hold at Kiowa on the 125° radial, right turns, 10 mile legs, expected further clearance time *[time now plus 15 minutes]*" ***[INSTALL HOLD; INSTALL HOLD PARAMS]****.*
Upon entering the hold: "[Company aircraft] 123, traffic has cleared out at Colorado Springs; you're cleared Kiowa, Victor 83 now" ***[EXIT HOLD]****.*
As aircraft is headed on course to Kiowa: "*[Company aircraft]* 123, intercept Victor 83" ***[INTERCEPT LEG-TO V83 (off route)]****.*
If PNF [subject] does not request it, the instructor delivers the ATIS, which informs the crew which runways are in use at the destination and provides information on weather ***[INSTALL APPROACH (ILS to Runway 17R)]****.*
Finally, the clearance to the ILS is delivered: "*[Company* aircraft] 123, cleared ILS 17R, Colorado Springs."
After receiving this clearance to the final approach course, the instructor – in the usual manner of a PF – gives the PNF this directive: "Extend the final approach course," or "Extend LACKI." ***[INTERCEPT LEG-TO FINAL APPROACH COURSE (off-route)]***

Figure 7. Transfer test in full-motion simulator

Session II that presented a flexible (not single-path) instructional program. These decisions about equipment and software were dictated in order to adhere to the guidelines for designing an intelligent tutor, particularly the guidelines to have students learn by solving problems, to promote progress to real-world performance and to provide appropriate error feedback (see Sections 2.2.5 and 2.2.6 above).

1.5 Procedures for training program

All experimentally trained pilots were in training for *no more than five hours* divided into two sessions. One of the authors (S.M.) served as the instructor for both training sessions and also administered the transfer tests after completion of the training. The instructor (S.M.) had worked on constructing the NGOMSL and had programmed the computer-based training (CBT) components in Authorware Professional™ for Session I and in SuperCard™ for Session II. The time lapse between Sessions I and II ranged from four days to several weeks.

After an introduction to the device and its basic operations, the experimentally trained pilots learned to perform the **Preflight FMC** CDU task. Then they learned how to carry out eight **Modify Route** CDU tasks (inflight tasks) embedded in realistic ATC clearances. During Session I the instructor-experimenter (S.M.) was the only other person present in the room and did not interact with the pilot in training once the pilot had begun using the CBT programmed in Authorware Professional™, except to notify the pilot of an appropriate break time.

The Session II training program simulated the performance of an intelligent tutor with its combination of the SuperCard™ program (run on a Macintosh platform) and a human instructor who supplied the intelligence. The instructional program for Session II consisted of realistic flight scenarios that posed problems for the pilot to solve. After a pilot selected a route and then a scenario for any **Modify Route** task, the instructor/experimenter (S.M.) would read the first of four simulated ATC directives to modify the previously installed route. Complete details of this training program are provided elsewhere (Polson, Irving, & Irving, 1995, Appendix C), including a description of the scenarios used in Session II of the training (see Appendix of this paper for one of the scenarios).

4 RESULTS: COMPARING THREE PILOT GROUPS' PERFORMANCE ON TRANSFER TEST

This section compares the performance of the three different groups of pilots on the transfer test in the full-motion simulator. We first analyzed the videotapes of each pilot's performance in the full-motion simulator, measuring performance times and classifying and recording all of the various specific errors committed and difficulties encountered by pilots while completing each of the 14 CDU tasks. We then assessed overall performance during the transfer test with two different quantitative measures: (1) an accuracy measure, the percentage of individuals in the group that were able to perform the task without any hints, and (2) a solution time measure, the time to perform each of the 14 CDU tasks. The accuracy measure is the better overall measure of performance, because it reflects the ability of the pilot both to perform the task independently and to perform the task within realistic time constraints. When the pilot was not performing the task rapidly enough to keep pace with the progress of the flight in the full-motion simulator, the experimenter was forced to intervene by giving a hint(s), thereby reducing the percentage of pilots who performed the task without hints.

Section 4.1 reports the between-group differences in the accuracy measure, and Section 4.2 reports between-group differences in the mean time to perform each task. Section 4.3 then closely analyzes determinants of performance deficiencies in the accuracy measure, focusing on understanding the performance deficiencies that were exhibited by all three pilot groups. The analysis traces performance deficiencies to three classes of determinants: differences in task difficulty, flaws in the CDU interface, and training design flaws. Section 4.4 analyzes performance deficiencies in the time required to perform each task by constructing a statistical model that explains most of the variance in performance of the three pilot groups on the transfer test. Section 4.5 then reports a follow-up experiment to test how to equalize the time-per-keystroke measure across the experimentally and traditionally trained.

4.1 Between-group differences in accuracy

Table 1 summarizes the results for the accuracy measure, the percentage that were able to complete the CDU tasks without hints from the experimenter. Table 1 reveals that preflight tasks were, on the whole, markedly easier than inflight tasks, and, as expected, the experts performed the best on both preflight and inflight tasks. Figure 8 shows the same data in greater detail, displaying the percentage of pilots in each of the three groups who were able to complete each of the 14 CDU tasks without hints. The tasks are arranged in the order they were performed during the transfer test. Pilots performed the six preflight tasks first (left side of Figure 8) and then performed the eight

Table 1. Mean percentage of pilots who completed CDU tasks without experimenter intervention: experts versus experimentally and traditionally trained groups

Groups of tasks	Experimentally trained (n = 19)	Traditionally trained (n = 19)	Experts (n = 10)	Mean all three groups (n = 48)
Preflight FMC (6 tasks)	84%	93%	100%	91%
Inflight modify route tasks (8 tasks)	64%	71%	80%	70%
Mean all 14 tasks	73%	81%	89%	80%

inflight tasks (right side of Figure 8).

Performance on eight of the 14 CDU tasks is both very good and very similar for all three groups, suggesting that the experimental training was successful for teaching the basic methods for manipulating the CDU. On these eight tasks the experimentally trained group performed slightly

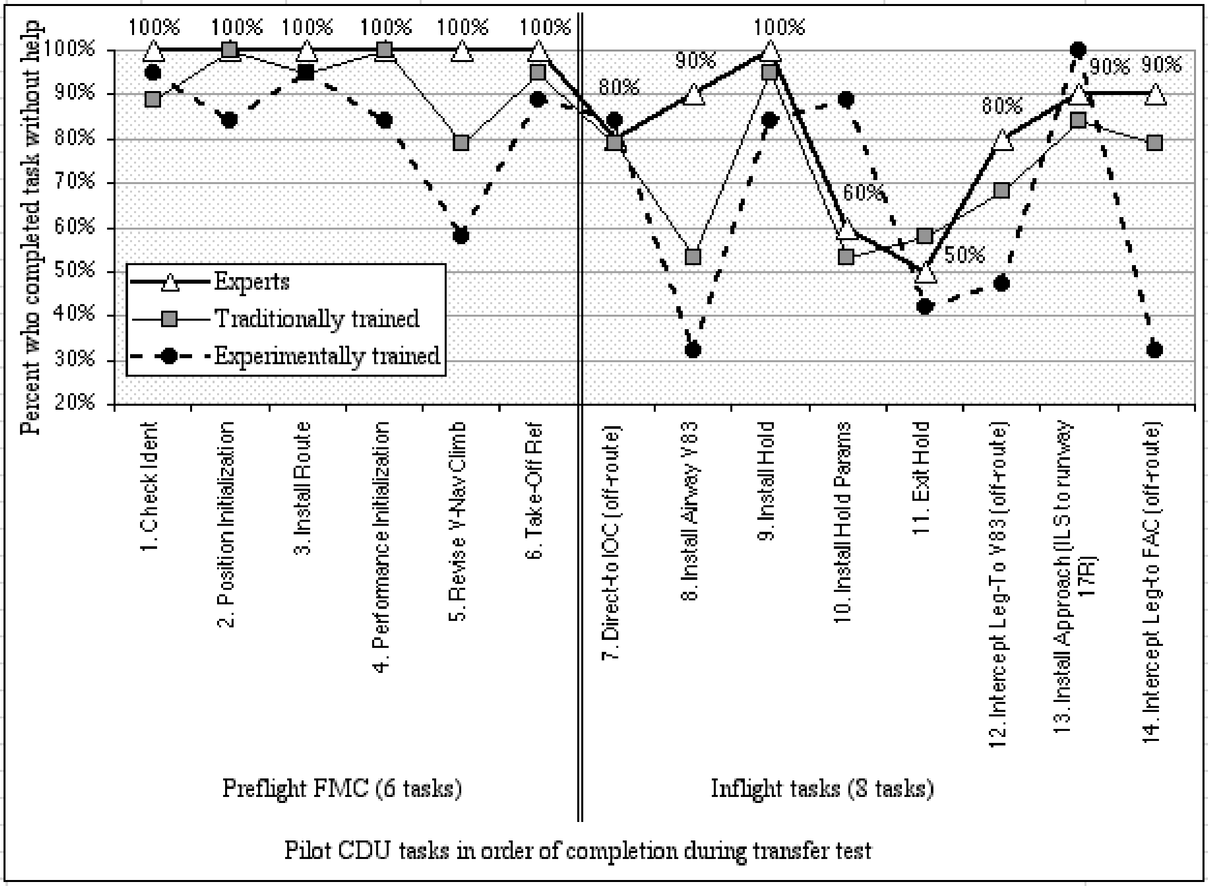

Figure 8. Percent who completed task without help: Experts vs. traditionally and experimentally trained

better than the traditionally trained on four tasks and slightly worse on the other four tasks. Chi Square tests found that none of the differences were statistically significant.

All individuals could carry out all 14 tasks, but, as Figure 8 shows, a significant fraction of pilots in all three groups required hints on the more difficult tasks. In Figure 8 the traditionally trained group appears to have performed better than the experimentally trained group on five of the harder tasks but their superiority was statistically significant only for the inflight INTERCEPT LEG-TO FINAL APPROACH COURSE task, χ^2 (1) = 8.622, p < .005. Balancing this, the experimentally trained group performed 36 percentage points better than the traditionally trained group on the INSTALL HOLD PARAMS subtask, χ^2 (1) = 6.269, p < .05. For the accuracy measure (the percentage of pilots who needed hints), therefore, the performance of the experimentally trained group was equivalent to the performance of the traditionally trained group.

1.2 Between-group differences in mean time to perform CDU tasks

Although Section 4.1 found no statistically significant differences between the experimentally trained and traditionally trained groups in the accuracy measure, the most important overall measure of performance, there is nevertheless a very clear pattern of performance differences in the time measure. A repeated measures ANOVA of the time per task shows significant main effects for group, F (2, 45) = 62.176, p <.0001, and for the sequence of 14 CDU tasks completed by each pilot, F (13, 585) = 31.633, p <.0001. The main effect for tasks reflects the fact that some tasks took much longer to perform than others, but the quantities and rankings of the between-group differences varied among the 14 CDU tasks, creating a significant interaction between group and task sequence, F (26, 5895) = 4.051, p <.0001.

Examining the data more closely, the experts – as expected because they had practiced the tasks more than any other group – performed the tasks the most rapidly. The mean time per task was 16.7 seconds for the experts, 22.0 seconds for the traditionally trained, and 39.9 seconds for the experimentally trained. Although the experts performed the tasks 5.3 seconds faster than the traditionally trained group, the Bonferonni/Dunn post hoc test showed that the difference was not significant. The experimentally trained pilots, however, took a mean of 17.9 seconds longer to perform each task than the traditionally trained pilots and 23.2 seconds longer than the experts, and Bonferonni/Dunn post hoc tests showed that these differences were statistically significant (both p <.0001).

We transformed the time data by dividing the time each of the 48 participating pilots took to complete each of the 14 CDU tasks by the minimum number of keystrokes required to complete the same task – shown in Table 2. The resulting time-per-keystroke measure is a mixture of the time to perform the physical keystroke and the mental operations needed to produce the action.

Figure 9 shows the mean time per keystroke taken by each group on each of the 14 CDU tasks. Time per keystroke should be relatively stable across tasks of equivalent difficulty, so graphing the time-per-keystroke measure for each group highlights both between-group and between-task differences that demand further analysis.[1] A repeated-measures ANOVA of the time-per-keystroke data shown in Figure 9 demonstrated a reliable main effect for group, F (2, 45) = 66.769, p <.0001, a

Table 2. Total keystrokes to perform each of 14 CDU tasks

CDU task listed in order completed	Preflight or inflight task	Total keystrokes
1. CHECK IDENT	Preflight	2
2. POSITION INITIALIZATION	Preflight	12
3. INSTALL ROUTE	Preflight	10
4. PERFORMANCE INITIALIZATION	Preflight	18
5. TAKE-OFF REF	Preflight	5
6. REVISE V-NAV CLIMB	Preflight	15
7. DIRECT-TO IOC (off route)	Inflight	6
8. INSTALL AIRWAY V83	Inflight	10
9. INSTALL HOLD	Inflight	6
10. INSTALL HOLD PARAMS	Inflight	13
11. EXIT HOLD	Inflight	3
12. INTERCEPT LEG-TO V83 (off route)	Inflight	4
13. INSTALL APPROACH (ILS to Runway 17R)	Inflight	5
14. INTERCEPT LEG-TO FINAL APPROACH COURSE (off route)	Inflight	4

reliable main effect for the sequence of 14 CDU tasks, F (13, 585) = 25.346, p <.0001, and a significant interaction between group and sequence of tasks, F (26, 585) = 4.969, p <.0001.

The ANOVA offers confirmation for what the eye sees in Figure 9. One pattern visible in Figure 9 is that the line representing the expert group is relatively flat, suggesting that expert performance is determined primarily by the number of keystrokes required to perform the task and

[1] The times-per-keystroke would have been longer on the harder tasks if the instructor had not intervened fairly quickly with a hint whenever the pilot taking the transfer test floundered on a hard task. The plane keeps flying the route in the full-motion simulator, so the pilot is necessarily under time pressure to get the CDU task done correctly. This truncates both the solution times and the times-per-keystroke for the harder tasks for all pilots who needed hints to perform the task.

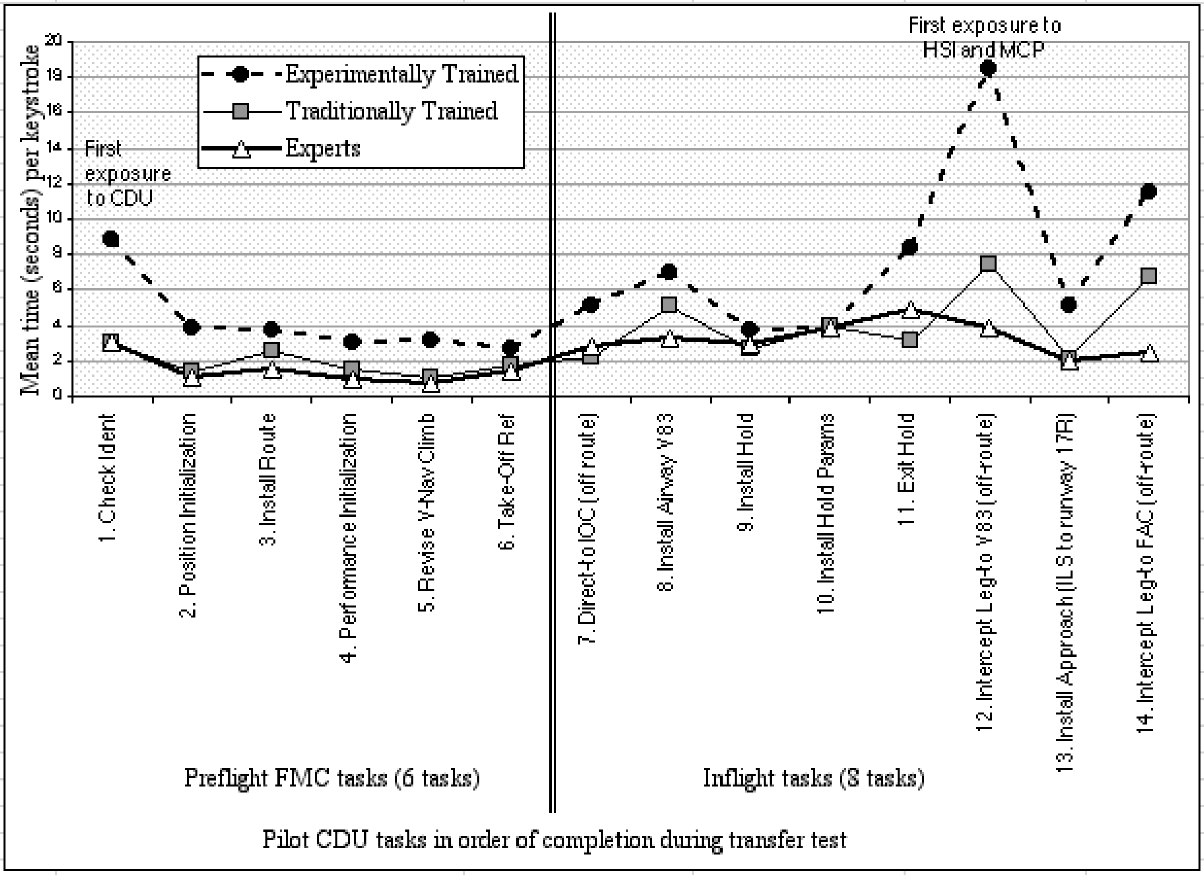

Figure 9. Mean time per keystroke to perform CDU tasks: Experts vs. experimentally and traditionally trained

their associated mental operations. The second visible pattern is that the time-per-keystroke measure is consistently higher for the experimentally trained group than for the expert group, while the time-per-keystroke of the traditionally trained group is close to the expert group. Third, the time-per-keystroke of all three groups are tightly clustered on the preflight tasks and the easier inflight tasks, but large gaps open up on four of the five tasks with known sources of difficulty. Fourth, the time-per-keystroke spikes sharply for the experimentally trained group on two tasks: CHECK IDENT and INTERCEPT LEG-TO V83.

The visible patterns in the time-per-keystroke measure in Figure 9 will be analyzed in detail in Sections 4.3-4.5, but there are no surprises in these patterns. There were several reasons to expect that the experimentally trained pilots would perform the tasks markedly slower than the traditionally trained pilots did. First, instructors of the traditionally trained pilots estimate that traditionally trained pilots had 10 to 50 hours of practice doing preflight and inflight tasks. In contrast, the experimentally trained pilots all completed the training in five hours or less. Thus, traditionally trained pilots spent up to ten times as much practice time as the five hours maximum total time the experimentally trained pilots spent completing the entire training sequence. Second, the experimentally trained group had no practice on the actual flight hardware, whereas the traditionally trained group practiced on an actual CDU. This explains the spikes in time-per-keystroke for the experimentally trained group on the CHECK IDENT and INTERCEPT LEG-TO V83. Third, and less important, the time gap between training and testing was much shorter for the traditionally trained pilots, who were all tested in the 24-hour window between their FAA checkride and the line oriented flight training session required before getting any line experience as a 737-300 pilot. People gradually forget skills they are not practicing/using, especially if they have not practiced these skills extensively, and performance degrades according to a power function (Anderson & Schunn, 2000).

1.3 Understanding determinants of performance deficiencies afflicting all three pilot groups

It is evident in Figure 8 that performance on individual tasks differed sharply. The mean percentage of all 48 pilots able to complete the task without hints ranged from 97% on the INSTALL ROUTE task down to only 50% on the EXIT HOLD task – a range of 47 percentage points. All three groups exhibited performance deficiencies on some of the hard CDU tasks, and it is very important to understand why. Trainers and training designers must understand the sources of difficulty for these tasks and consider how to improve training so that expert pilots no longer make mistakes on these tasks.

In general performance deficiencies issue from three sources of difficulty. First, some tasks are inherently harder than others, and it is necessary to identify the exact conceptual difficulties and/or complexities in the procedures that explain the higher error rates for the harder tasks and that offer insights into how to improve training for these tasks. Second, there are flaws in the CDU interface that pilots must learn to compensate for. Flaws in the CDU interface cause task-mapping errors, forcing pilots to translate other sources of information (such as ATC directives) into a different form before entering the information into the CDU. Third, there are training design flaws. The experimental training, for example, did a better job of teaching pilots to compensate for CDU interface flaws on the INSTALL HOLD PARAMS inflight task but a poorer job of teaching pilots to compensate for the CDU interface flaw on the REVISE V-NAV CLIMB preflight task.

1.3.1 Preflight tasks

Preflight tasks are generally easy, because the CDU presents specific prompts for preflight tasks – primarily forms the pilot must fill out that take the pilot step by step through the six preflight tasks. For five of the six preflight tasks the mean percentage who completed the task without hints was 95%. There is, however, one outlier among the preflight tasks: REVISE V-NAV CLIMB. The percentage who completed REVISE V-NAV CLIMB without hints was only 79%, far lower than the mean for the other tasks. The NGOMSL model shows that the sequence of preflight tasks establishes a pattern of prompting the pilot to **ACCESS** the correct page to do the next task in the sequence. Pilots become accustomed to using an identical **ACCESS** method for the preflight tasks, but REVISE V-NAV CLIMB poses a sudden exception to the rule, causing a dip in performance for both the experimentally and traditionally trained groups. The pilot must recall that the REVISE V-NAV CLIMB is an exception to the rule, use the CLB function key (not a displayed prompt) to **ACCESS** the appropriate page, complete the action, and then return to the path (without a prompt) to do the next task in the sequence, TAKE-OFF REF.

Due to the additional line experience, experts had fully mastered the exception to the rule caused by this inconsistency in the CDU interface, and all ten experts performed REVISE V-NAV CLIMB without hints. Only 79% of the traditionally trained and 58% of the experimentally trained could perform the task without hints. The traditionally trained group had more practice on the tasks than the experimentally trained group and more closely approximated the perfect performance of the experts. In addition, the experimentally trained group may have performed temporarily worse because the experimental training facilitated transfer of training by focusing on the common intermediate-level goals, arousing a stronger expectation of consistency in the CDU interface and creating negative transfer effects for the one exception to the otherwise consistent pattern. Future iterations of the experimental training should retain the focus on the common intermediate-level goals but highlight the inconsistency flaw in the CDU interface and how to compensate for it. Experimental training should also provide more practice time on this task.

1.3.2 Inflight tasks

As Figure 8 shows, inflight tasks vary widely in difficulty, ranging from easy to very difficult. The mean percentages of all 48 pilots who completed the two easiest tasks, INSTALL HOLD and INSTALL APPROACH (ILS to Runway 17R) were 93% and 91%, respectively. At the other end of the continuum, the mean percentages for all 48 pilots on the five most difficult tasks ranged from 67% down to only 50%, averaging 61%. These five inflight tasks present formidable sources of difficulty that took a heavy toll on pilot performance.

The sources of difficulty in these five inflight tasks were heterogeneous, and some continue to afflict even experts with a year or more of line experience. On average, only 74% of the experts were able to complete these five tasks without hints, and expert performance dipped to 50% on the inflight EXIT HOLD task and 60% on the inflight INSTALL HOLD PARAMETERS subtask. The paragraphs that follow describe in detail the particular sources of difficulty for each of the five tasks.

Performance on INSTALL HOLD PARAMS is damaged by a task-mapping error caused by a discrepancy between the CDU user interface requirements and the ATC environment, requiring the pilot to translate the information before entering the information into the CDU scratchpad. The ATC clearance tells the pilot to hold at KIOWA on the 125° radial, which is calculated *from the waypoint* (KIOWA). The FMC, however, flies a course in the opposite direction, flying *to the waypoint*. Consequently the pilot must convert the 125° radial to its reciprocal (125° + 180° = 305°) and then **DESIGNATE** the reciprocal (305°) in the CDU. The pilot must also remember to do this conversion without being prompted, and, not surprisingly, many pilots incorrectly entered 125° instead of 305°. The percentages not needing hints for INSTALL HOLD PARAMS were only 60% for the experts and 53% for the traditionally trained, but much higher – 89% – for the experimentally trained pilots. In this instance the experimental training design anticipated the source of difficulty in advance and was especially successful in overcoming it.

EXIT HOLD presents another task-mapping difficulty. The first step in the NGOMSL method for EXIT HOLD is to determine if the ATC directive requires returning to the holding fix. The clearance given in the transfer test was, "You're cleared KIOWA, Victor 83, Colorado Springs." This technically means that the pilot must return to the holding fix, KIOWA (IOC), before continuing on with the programmed route. The Hold page provides the appropriate prompt. If the CDU is already displaying the Hold page, all the pilot needs to do is press the line select key at 6R, "> EXIT HOLD," and then press the Execute key. The next time the aircraft crosses the holding fix (IOC) the FMC will command the aircraft to depart holding and continue with the installed route (the V83 airway installed three tasks prior to this task).

The CDU is, however, more likely to be displaying the LEGS page, a typical page during flight. There is no cue to EXIT HOLD on the LEGS page and none in the ATC directive either. (Sometimes the ATC directive explicitly includes the words "exit hold," but more often it does not.) The typical pilot error is to translate the ATC directive into some kind of DIRECT-TO action to a waypoint down-path from KIOWA. The percentages not needing hints for EXIT HOLD were low for all three groups: 42% for the experimentally trained, 50% for the experts, and 58% for the traditionally trained. This suggests that the next iteration of the experimental training should further unpack the first step in the NGOMSL method for EXIT HOLD, explicitly teaching pilots exactly how to determine if the ATC directive requires returning to the holding fix and if EXIT HOLD is the proper method to comply with the ATC directive.

The two INTERCEPT LEG-TO (off route) tasks, INTERCEPT V83 and INTERCEPT FINAL APPROACH COURSE (FAC), each presented two sources of difficulty. The first source of difficulty comes from the fact that the CDU page for this task is shared with the DIRECT-TO task, and there are five positions for waypoints on this page (LEGS page). According to the NGOMSL model, this raises the probability of **INSERT** errors. If the pilot inserts the waypoint in the highly salient line position 1L, this will command the FMC to fly the plane *direct to that waypoint* and make the flight path out of compliance with the ATC directive. The second source of difficulty was the necessity of re-mapping the clearance to a CDU-compatible goal and using the HSI display to get the information needed for re-mapping. The pilot must re-map the clearance "INTERCEPT V83" to the CDU-compatible goal of "INTERCEPT the next down-path waypoint on V83." (Many pilots mistakenly inserted the airway, V83, instead of inserting the next down-path waypoint on V83.) Similarly, the pilot must re-map the clearance for INTERCEPT FINAL APPROACH COURSE by extending a fix on that approach course and giving the pilot graphic assistance from the HSI to guide the aircraft to the glide slope. The percentages not needing hints for the INTERCEPT LEG-TO V83 and INTERCEPT LEG-TO FINAL APPROACH COURSE (FAC) tasks were 47% (V83) and 32% (FAC) for the experimentally trained, 68% (V83) and 79% (FAC) for the traditionally trained, and 80% (V83) and 90% (FAC) for the experts. Even experts continued to have problems with INTERCEPT tasks, but the performance of the experimentally trained was especially poor (see Sections 4.2 and 4.3 for an explanation of the additional difficulties encountered by the experimentally trained).

The INSTALL AIRWAY V83 task posed two sources of difficulty. First, the pilot must remember a complex procedure. The correct page for entering airway designations is the Route page and pressing the RTE function key on the CDU is the way to **ACCESS** the Route page. In order to put the airway on the Route page, the crew must first establish a starting point for the airway. For example, in order to intercept the airway into the waypoint Kiowa (IOC), the fix before IOC would be entered into the Route page, then the airway V83 inserted into 1L or 2L for IOC. This would put the correct radial on the LEGS page and allow the crew to intercept it. Finding the fix before IOC would require the crew to recall it from memory (doubtful) or look it up on a paper chart or the HSI. Second, almost all modifications to the lateral routing are made on the LEGS page, where waypoints are entered, but airways are an exception. The pilot must remember that the CDU will not accept an airway designation attempted on the LEGS page, countering the frequency bias toward entering route elements on the LEGS page. The CDU responds to an attempt to enter an airway on the LEGS page with a cryptic error message, "Not in database." A pilot was liable to interpret this error message as indicating a typographical error, blocking recognition of the real problem: that the pilot had failed to **ACCESS** the correct page (the Route page). The percentages not needing hints for INSTALL AIRWAY V83 were 90% for the experts but only 53% for the traditionally trained and 32% for the experimentally trained.

The sources of difficulty are summarized in Table 3. A simple regression analysis shows that a single independent variable, the number of sources of difficulty, explains 77% of the between-task variance in the accuracy measure (the mean percentages of the experimentally and traditionally trained groups who completed the various CDU tasks without hints). The number of sources of difficulty for this analysis was zero for each of the 14 tasks not discussed in this section, one each for the EXIT HOLD and INSTALL HOLD PARAMS tasks, and two each for INSTALL AIRWAY V83, INTERCEPT LEG-TO V83, and INTERCEPT LEG-TO FINAL APPROACH COURSE. For a parallel analysis expanded to include all three groups of pilots, the number of sources of difficulty variable explained 68% of the between-task variance in task difficulty. In sum, the sources of difficulty identified in this section account for most of the variance in the number of hints pilots needed to complete the various CDU tasks.

A very valuable benefit of doing a fine-grained cognitive task analysis model, such as NGOMSL, is that it makes it possible to pinpoint and explain between-task differences in difficulty and then use the information gained to improve the next iteration of training for the most difficult tasks. A NGOMSL model provides very detailed steps, both physical and mental operations necessary to carry out the CDU tasks. From our first-iteration fine-grained NGOMSL model we have been able to derive explanations of what causes performance problems, including conceptual difficulties and/or procedural complexities, flaws in the interface that cause frequent task-mapping errors, and training design flaws. Pinpointing the particular steps in these tasks that caused errors or hindered performance has, in addition, revealed aspects where the first-iteration model was incomplete. It is important to further decompose these troublesome steps, making the NGOMSL model more complete. A more complete NGOSML model, in turn, would lay the foundation for designing a more effective curriculum for future iterations of the CDU training, more effectively teaching pilots how to overcome these sources of difficulty. The goal is to ensure that expert pilots would quickly reach

Table 3. Sources of difficulties in CDU tasks during transition training to advanced automation aircraft

Task	Number	Description of specific difficulties
INSTALL AIRWAY (V83)	2	1) Pilot must remember a complex procedure: (a) **ACCESS** Route page, (b) **DESIGNATE** starting waypoint for the airway by retrieving it from long term memory or looking it up, (c) **INSERT** the fix before the starting waypoint on the Route page, and (d) **INSERT** the airway. 2) Modifying a route by adding an airway is an exception to the usual rule – all other route modifications are performed on LEGS page but airways must be added on RTE page. If pilot enters airway in waypoint position the pilot is not apt to recover from the error, because CDU returns a cryptic error message that seems to indicate typographical error.
INSTALL HOLD PARAMS	1	1) Without any cue, must recognize that FMC calculates from waypoint to aircraft (unlike ATC and pilots, who calculate from aircraft to waypoint) and hence convert radial in ATC directive to its reciprocal and enter reciprocal in CDU.
EXIT HOLD	1	1) Neither ATC directive nor LEGS page (usual page during flight) provides a prompt to return to the holding fix, but must use EXIT HOLD to ensure that the aircraft crosses over the holding fix before continuing on with the installed route.
INTERCEPT LEG-TO V83 (off route)	2	1) CDU page for INTERCEPT shared with DIRECT-TO, forcing conceptual distinction that requires entering data into less salient CDU line positions on shared page. 2) Must re-map clearance to CDU-compatible goal and use HSI display to get information needed for re-mapping.
INTERCEPT LEG-TO FINAL APPROACH COURSE (off route)	2	1) CDU page for INTERCEPT shared with DIRECT-TO, forcing conceptual distinction that requires entering data into less salient CDU line positions on shared page. 2) Must re-map clearance to CDU-compatible goal and use HSI display to get information needed for re-mapping.

excellent levels of performance on all the difficult CDU tasks, not just the easier ones. The first-iteration NGOMSL model also identified key flaws in the CDU interface, so a complete NGOMSL model would make it possible to improve future iterations of the CDU interface design.

1.4 Model of determinants of performance deficiencies in the time measure

Section 4.3 looked at performance deficiencies that affect all three pilot groups, showing that the number of sources of difficulty explains most of the *between-task* differences in performance on the accuracy measure. Section 4.4 shifts attention to constructing a model of *between-group* differences in performance. This section focuses on explaining the statistically significant differences in performance on the time to perform each task (see above, Section 4.2). In the accuracy measure there were no significant overall *between-group* differences (see above, Section 4.1).

To better understand Figure 9 and the repeated-measures ANOVA of the time per task data, we constructed and tested two different multiple regression models of the determinants of mean solution time for each task. The first multiple regression model used two independent variables expected to have similar effects on both the experimentally and traditional trained groups. The first common determinant of mean task solution time is the minimum number of keystrokes required to complete the task (see Table 2), assuming that task solution time is roughly proportionate to the number of keystrokes. The second common determinant is the number of known sources of difficulty during training. Table 3 (above) summarizes the sources of difficulty for inflight tasks that were discussed in detail in Section 4.3.2. In general, these sources of difficulty should have slowed down both experimentally and traditionally trained pilots' performance on the more difficult tasks.

The second multiple regression model retains the same two common independent variables but adds a third variable scored for each particular CDU task: the number of hardware devices that experimentally trained pilots encountered for the first time during the transfer test. The preflight CHECK IDENT task was coded as one, the inflight INTERCEPT LEG-TO V83 task was coded as two, and the remaining 12 CDU tasks were coded as zero because they provided no exposure to new hardware devices. CHECK IDENT was the first task performed during the transfer test, exposing the experimentally trained group for the first time to the actual CDU hardware. Experimentally trained pilots had performed all training tasks using a mouse to punch keys on a graphic image of the CDU displayed on the monitor, and pilots in the experimental group had no experience doing keystrokes on a real CDU device until the transfer test in the full-motion simulator. The inflight INTERCEPT LEG-TO V83 task was the first of the two INTERCEPT LEG-TO tasks to be performed. The INTERCEPT LEG-TO V83 task gave experimentally trained pilots their first exposure to coordinating the CDU interface with two additional hardware devices, adjusting the knob on the mode control panel (MCP) and viewing the change in routing on the horizontal situation indicator (HSI).

We hypothesized that this third variable (number of devices first encountered) would have a strong effect on the experimentally group but no effect on either the experts, who were very familiar with the flight hardware from line experience, or the traditionally trained group, who practiced extensively with the actual flight hardware during their transition training. Figure 9 suggests that the hypothesis is correct, showing a spike on first exposure to the hardware devices only for the experimentally trained group but not for the other two groups.

The multiple regression models, however, provide a real test of the hypothesis. Table 4 displays the coefficients, expressed as number of seconds, from all four multiple regression analyses, two that test the two-variable model on each of the training groups and two that test the three-variable model. The two-determinant model explains 82% of the variance for the experimentally trained group and

Table 4. Summary of 4 multiple regression analyses that explain mean total time per task for the experimentally and traditionally trained groups as a function of either 2 or 3 common independent variables

Variable	Coefficient for variable expressed in number of seconds			
	Experimentally trained		Traditionally trained	
	2-variable analysis	3-variable analysis	2-variable analysis	3-variable analysis
Intercept	10.584 n.s.	***5.810 n.s.***	***-.207 n.s.***	.319 n.s.
Total keystrokes	2.398 **	***2.837 *****	***1.824 *****	1.776 *
Number known sources of difficulty for training	17.431 **	***15.154 *****	***13.128 *****	13.378 **
First exposure to each hardware device during transfer test	–	***11.807 *****	–	-1.301 n.s.
R^2 values (proportion of variance explained)	$R^2 = .82$	***$R^2 = .92$***	***$R^2 = .75$***	$R^2 = .76$

Note: Columns in boldface italics show the best analysis for each group

* *p* < .01

** *p* <.005

n.s. not statistically significant (*p* > .05)

75% of the variance for the traditionally trained group. As expected, the third determinant is statistically significant for the experimentally trained group, and the three-determinant model explains 92% of the variance and is the best model for the transfer test data from the experimentally trained group. For the traditionally trained group, in contrast, adding the third determinant fails to increase the percentage of variance explained, and the third determinant is not statistically significant and should be deleted for the traditionally trained group. Therefore, as expected, the two-determinant model provides the best explanation of the data for the traditionally trained group.

Thus, the center two columns of Table 4 – marked by boldface italics – highlight the best multiple regression model for each training group, the experimentally trained and the traditionally trained. Comparing the coefficients in these two columns pinpoints the effects of these variables on each group. The experimentally trained group took 2.8 seconds per keystroke, 56% longer than for the traditionally trained group (1.8 seconds). The experimentally trained group spent 15.2 seconds to solve each conceptual difficulty, 15% longer than for the traditionally trained group (13.1 seconds). The first exposure to the hardware adds 11.8 seconds per hardware device, meaning 11.8 seconds for the CHECK IDENT task (5.9 seconds extra per keystroke for this two-keystroke task) and 23.6 seconds for the INTERCEPT LEG-TO V83 task (5.9 seconds extra per keystroke for this four-keystroke task). The intercept is not statistically significant for either group.

The statistical analyses shown in Figure 9 and the multiple regression models shown in Table 4 match qualitative observations by the author (S.M.) who supervised the transfer test in the full-motion simulator and analyzed the videotape records of the test. Experimentally trained pilots rapidly became comfortable with operating the actual CDU but found the novel HSI both more challenging and more interesting. Prior to the transfer test the experimentally trained pilots had relied on paper charts for checking the route of flight. Paper charts were, however, not available in the simulator,[1] so the HSI display was the only source of map information available to pilots during the transfer test, e.g., to see that Kiowa (IOC) was, indeed, on airway V83.

When the directive came to INTERCEPT V83 the experimentally trained pilot for the first time used the HSI and MCP to complete a CDU task. The HSI (a "moving map" display that superimposes the horizontal flight path on an electronic map) assists the flightcrew in visualizing and understanding the implications of clearances. Experimentally trained pilots did not have access to the actual HSI and MCP flight hardware during training – not until the transfer test – and did not even have access to a faithful approximation of that hardware. They had seen only a simplified static representation of the HSI. During the first session of training the task was explained via crude, static images. Even during the second session of training, the experimentally trained pilots could not manipulate the mode control knob and get the realistic feedback that would have been available from an actual HSI. Furthermore, using the HSI to pick some down-path waypoint to complete the INTERCEPT V83 task presupposed first learning how to comprehend the HSI display as a source of map information. Therefore, the HSI "moving map" display was new to these pilots in the transfer test. Fortunately, somewhat earlier in the transfer test (before attempting the INTERCEPT V83 task) each experimentally trained pilot had had a few moments to satisfy his/her curiosity about the novel HSI in the simulator and explore the "moving map" display, observing the map's image of the airplane traversing the route (depicted as a bright magenta line). Trying to make sense of that moving map displaying the "magenta line," and perhaps verifying their position vis-a-vis the route displayed on the ROUTE or LEGS page, posed a significant level of fascination and challenge. Thus, the demands of the transfer test prompted experimentally trained pilots – in the brief amount of time available – to both learn to interpret the information on the HSI display and learn how to use the HSI and MCP to complete the INTERCEPT V83 task.

[1] In all "glass-cockpit"/FMC aircraft paper charts always take priority over the onboard display, so many pilots politely protested not having access to paper charts during the full-motion simulator transfer test. The purpose of not making paper charts available was to force pilots to rely on the HSI.

Not surprisingly, inadequate representations of the HSI and MCP appeared to exacerbate the accuracy measure of performance on the INTERCEPT tasks as well as the time-per-keystroke measure. The INTERCEPT LEG-TO FINAL APPROACH COURSE task was the only task on which the experimentally trained group performed significantly worse on the accuracy measure than the traditionally trained group (see Section 4.1). The experimentally trained pilots also performed poorly on the accuracy measure for the other INTERCEPT LEG-TO task, INTERCEPT V83, although the difference was not statistically significant. The use of an inadequate part-task simulator proved to be a glaring failure in the experimental training design, contributing to this poor performance on the two INTERCEPT tasks. Clearly, the next iteration of the experimental training must correct this training deficiency by either providing practice with an actual HSI and MCP or by providing a more faithful approximation of the HSI that changes realistically in response to rotations of the mode control knob.

Although developing a fine-grained model of performance, such as our NGOMSL model, provides the content for the training design (see the description of the first principle for designing intelligent tutors, Section 2.1) it is equally necessary to effectively deliver that content (see the other seven principles for designing intelligent tutors, Sections 2.2.1 to 2.2.7). The second iteration of training would improve both the NGOMSL model and the delivery of content to reap gains in performance on those tasks that revealed serious sources of difficulty.

1.5 Follow-up experiment on repairing performance deficiencies in the time measure

As we noted in the previous section, the one outlier for the **Preflight FMC** tasks was the performance of the experimentally trained pilots on CHECK IDENT, because this task was the experimental training pilot's first attempt to use the flight hardware CDU keyboard. The single time spike on the CHECK IDENT task suggests the experimentally trained pilots could quickly transfer their skills learned on a Macintosh to the actual flight hardware and partially adapt to the new input medium. Their higher time-per-keystroke performance on the rest of the preflight and inflight tasks, however, shows that they consistently performed the keyboarding skills considerably slower than the traditionally trained pilots did. In this section we turn our attention to deepening the analysis of how well part-task training transferred to the full-motion simulator, focusing on the consistently slower time-per-keystroke performance of the experimentally trained pilots and how to improve it.

Traditionally trained pilots had spent 10 to 50 hours practicing in a fixed-based simulator. Experimentally trained pilots clearly needed more practice time to lower the time-per-keystroke measure, but the question here is specifically whether they also needed practice in a simulator. An additional advantage of spending time in a fixed-based simulator would be to eliminate the spikes in time-per-keystroke for first exposure to the flight hardware. Our explanation of the time-per-keystroke differences between the traditionally and experimentally trained pilots has focused on the experimentally trained pilots having to adapt to the CDU keyboard and learn to use the MCP and understand the interaction between the MCP and the HSI. We evaluated these claims in a follow-up experiment that administered the five-hour experimental training but substituted some practice in a fixed-based simulator. This gave the simulator group one and one-half hours of experience performing the same CDU tasks with the actual flight hardware, compared to the experimentally trained pilots who practiced the CDU tasks using only a primitive simulation of the HSI and MCP.

This "simulator" group ($n = 6$) was comprised of undergraduates in an aerospace science program who aspired to become commercial pilots. The "simulator" group performed the same version of the Session I experimental training as the experimentally trained group did. For the Session II training, however (using the same realistic flights and ATC directives to modify the installed route), the "simulator" group carried out the tasks in a fixed-base simulator (rather than on the SuperCard™ tutor used by the experimentally trained group). The amount of time for Session II in each case was kept the same—about 80-90 minutes. The "simulator" group then completed the same transfer test in the full motion simulator. A repeated-measures ANOVA comparing mean time per task shows no significant difference between the two groups, $F(1, 23) = 3.815$, $p = .063$, and no significant group-

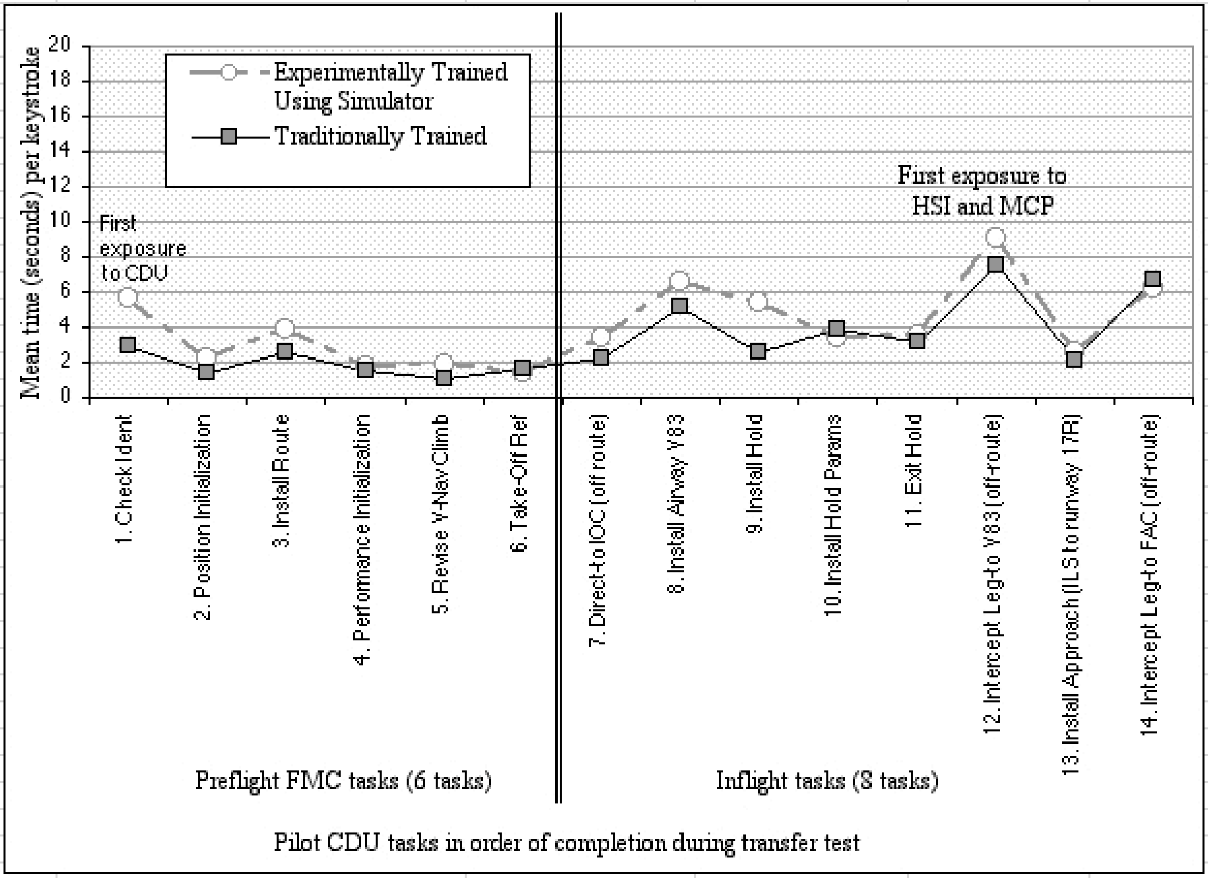

Figure 10. Comparison of mean-time-per-keystroke for experimentally trained pilots that used a simulator and traditionally trained pilots

by-tasks interaction, F (13, 299) = 0.539, p = .899. Figure 10 provides a visual summary of these data by first converting the time-per-task data for the “simulator” group to time-per-keystroke data and then comparing the time-per-keystroke data for the "simulator" group and traditionally trained group (cf. Fig. 9, Section 4.2).

As expected, a multiple regression analysis of mean task time versus two independent variables – total keystrokes and number of known sources of difficulty – found that both variables were significant (p <.05) and together accounted for a large percentage of the variance for the "simulator" group (R^2 = .63). Adding the third variable – first exposure to each hardware device during the transfer test – had no effect on the R^2 value of the multiple regression analysis, and the third variable was not significant. Therefore, the two-variable multiple regression model is the best model for the “simulator” group, just as it was for the traditionally trained group (cf. Table 4, Section 4.4). Furthermore, the coefficients for the two-variable multiple regression model showed that the “simulator” group took 2.0 seconds per keystroke, much less time than the 2.8 seconds taken by the experimentally trained group and only 11% more than the 1.8 seconds per keystroke taken by the traditionally trained group.

These data strongly suggest that as little as 80-90 minutes of practice (or less) in a fixed-base simulator with the actual FMC hardware could be sufficient to erase the effects of first exposure to the actual hardware devices during the transfer test for the experimentally trained group. Alternatively, similar results could probably be achieved with some practice on modern desktop/laptop simulators with excellent simulations of the Navigation Display (ND) and MCP and provisions for lesson plans. Such enhanced simulations improve understanding of the integration of

CDU, HSI, and MCP for carrying out the INTERCEPT tasks in our study (more generally, the integration of the CDU, ND, and MCP).

5 Conclusions

Educational programs can claim to be more effective if they result in a higher level of learner performance while holding learning time and training costs constant, or if they result in the same level of learner performance while reducing learning time and/or cost of training equipment. One major goal of our research, discussed in Section 5.1, was to provide proof of concept for the latter claim to educational effectiveness: enabling learners to achieve the same level of CDU performance that is achieved in traditional training while reducing costs for training equipment and training time. Accomplishing this goal would, in turn, free resources to achieve the other claim for greater educational effectiveness: actually raising the performance level while holding constant the resources invested in training. Flightcrews need to be trained in a much larger repertoire of CDU skills and to gain a better understanding of the FMC, but it is not feasible to increase the training time. The higher performance level on the flightcrew-automation interface must derive from greater educational effectiveness that enables covering an expanded curriculum while holding constant the training time and costs.

The second major goal of our research, discussed in Section 5.2, was to design more effective aviation training based on the ACT-R skill acquisition theory and the eight principles for designing ACT-R intelligent tutors. Section 5.2 evaluates the potential value of applying the training design developed in this research project to designing other more effective aviation training programs, including – but not limited to – training programs for operating the flightcrew-automation interface. Of particular note, further gains in educational effectiveness could be gleaned from joining future iterations of this training design (based on ACT-R) with current-generation desktop/laptop simulators that offer provisions for lesson plans (e.g., Aerosim, 2001b).

5.1 Delivering same performance results with more time- and cost-efficient training

The short-term aim of our experimental training program was to demonstrate the feasibility of using less expensive training equipment and reducing learning time without reducing the resulting level of performance (a level that matches or exceeds the standard set by the traditionally trained pilots). The ideal training design would have been to set the desired performance criteria in advance and then require each and every pilot to invest the amount of total training time necessary to reach the performance criteria. Experimental participant availability constraints, however, forced us to compromise on that ideal. To obtain enough pilots to participate in our experiment, we designed the training so that it could be completed in five hours – approximately one-sixth as much time as traditionally trained pilots invest, on average, in practicing the CDU tasks during their five-week transition training program for piloting FMC-equipped aircraft. In fact, all experimentally trained pilots did complete the training in five hours or less.

The overall outcome of the transfer test, as we saw in Section 4, was that the experimentally trained pilots approximated the performance of the traditionally trained pilots in the accuracy measure but were significantly slower in performing the tasks. Section 5.1.1 evaluates whether the training experiment provides proof-of-concept for the claim of greater educational effectiveness in relation to the reduced cost of training equipment. Section 5.1.2 evaluates whether the training experiment provides proof-of-concept for the claim of greater educational effectiveness in relation to the reduced length of training time.

5.1.1 Effectiveness of part-task training to minimize equipment costs for CDU training

The first iteration achieved equivalent results in accuracy measures using part-task training on inexpensive equipment, but there was one particularly serious problem associated with part-task training. Significantly fewer pilots in the experimentally trained group could complete the

INTERCEPT LEG-TO FINAL APPROACH COURSE (FAC) task without hints, and their mean times per keystroke were disproportionately high on both the INTERCEPT V83 and INTERCEPT FAC tasks (18.5 and 11.5 seconds for the experimentally trained versus 7.5 and 6.7 seconds for the traditionally trained group). The disparity in performance on INTERCEPT tasks can be explained by the fact that the experimentally trained group had no exposure to the actual flight hardware and not even an adequate simulation of the HSI and MCP flight hardware during practice on the INTERCEPT tasks[1] (see discussion above, Section 4.4). The next iteration of experimental training needs to expose pilots to the actual HSI and MCP hardware and/or to adequate simulations of the actual flight hardware devices.

Figure 9 and the multiple regression analyses in Table 4 together suggest that the experimentally trained group adjusted rapidly to the new hardware devices. Although the experimentally trained group exhibited longer per-keystroke times when they first encountered the actual flight hardware, these effects were spikes in time limited to the single task on which they were first exposed to the new hardware. This implies that transfer test performance of experimentally trained pilots would have benefited from at least some prior exposure to the actual flight hardware. The additional "simulator" group experiment verified this interpretation by eliminating the time spikes (verified by the results of the two- and three-variable multiple regression analyses).

In contrast to the primitive desktop simulator used by the experimentally trained pilots, modern desktop/laptop simulators (e.g., Aerosim, 2001a, 2001b; Tricom Technologies, 2001; Wicat, 2001a, 2001b) offer excellent simulations of the HSI and MCP, as well as improved simulations of the CDU. If experimentally trained pilots had been able to do the Session II flight scenarios on current generation desktop/laptop simulators, it would probably have eliminated the spikes in time that the experimentally trained group manifested on first exposure to the hardware devices. It is worth testing this possibility, because it could obviate the necessity for any practice time in a fixed-base or full-motion simulator during training.

In sum, the experiment provides strong evidence that skills acquired through part-task training do enable pilots to adequately perform the CDU tasks that are tested on the FAA checkride. The experimentally trained group very rapidly adjusted after their first exposure to the actual flight hardware in the full-motion simulator, so skills learned in part-task training clearly do transfer effectively to a full-motion simulator. Our results are consistent with research by other investigators on transfer of various skills acquired from part-task training on low-fidelity devices (Dennis & Harris, 1998; Gopher, Weil, & Bareket, 1994; Jentsch & Bowers, 1998; Koonce & Bramble, 1998).

Results from our second experiment show that eliminating the spikes in time can be accomplished by including a short session of practice in a fixed-base or full-motion simulator. It can probably be accomplished alternatively with some practice on modern desktop/laptop simulators with excellent simulations of the HSI and MCP and provisions for lesson plans. In addition to improving performance on INTERCEPT tasks, this exposure would increase pilots' familiarity with the way the CDU, HSI, and MCP interact, facilitating an understanding of and confidence in the FMC.

5.1.2 Delivering training effectively to substantially reduce training time

In general, mature ACT-R tutors are designed to optimize learning rate, and students using ACT-R tutors typically reach "at least the same level of mastery as control students in about one third of the time" (Anderson 1993, p. 236). ACT-R tutors offer two main sources of training time reductions: (1) immediate error feedback, and (2) transfer of training. Our experimental training used both. Immediate error feedback in Session I guided learners as efficiently as possible toward acquiring competence with a deep level of understanding. As discussed in Section 2.2.6, mastery of a skill

[1] Our NGOMSL analysis had pointed out the need for the HSI in our training. Because of limitations on computer horsepower and programming difficulties, however, we chose to go without an adequate simulation of the HSI. As the NGOMSL analysis predicted, the experimentally trained pilots made many errors when first exposed to the HSI on the INTERCEPT LEG-TO V83 task.

results from acquisition of the condition-action rules required for competence, regardless of how long it takes to acquire these rules. Anderson and colleagues have shown that giving learners immediate error feedback greatly reduces the time learners waste floundering. In comparison, training conditions that give no feedback or less effective feedback consume two to three times as much training time, on average, without improving the end-result performance level or capacity to transfer to novel tasks.

Structuring training to facilitate transfer is another means we adopted to achieve training time reductions. Structuring the training around the common intermediate goals **ACCESS, DESIGNATE,** and **INSERT,** increased across-task transfer and avoided the necessity to teach each of the 14 CDU tasks as a sequence of separate keystrokes and mental operations. Nesting the eight inflight tasks under the higher-level goal **Modify Route** should also have increased transfer. We have no way of estimating the additional training time reductions that can be expected from facilitating transfer.

In addition to training time reductions from immediate error feedback and transfer of training, we reduced training time by using instructional software that jumped down the flight route to the waypoint for the next CDU problem, channeling virtually all of pilots' learning time to solving problems. In contrast, pilots being trained in fixed-base or full-motion simulators must wait for the simulator to fly to the waypoint for the next problem. We have no way of separating out the savings in training time that can be accrued through avoiding this “real time” problem of the fixed-base or full-motion simulator as a practice tool. Actually, some fixed-base simulators do offer the option of flying two or four times as fast as “real time,” reducing waiting (“en route”) time. But what is perhaps more important than the time lost waiting for the fixed-base or full-motion simulator to traverse the route in real time is the fact that pilots are distracted in these simulators. Pilots become more absorbed in pushing the throttles and flying the plane than they are in learning to operate the CDU. In contrast, the pilot in the part-task training remains fully focused on learning the CDU tasks. The fixed-base or full-motion simulator is appropriate for integrating skills already learned in part-task training environments, but it is inefficient for acquiring the fundamental condition-action rules for basic competence.

Extrapolating from Anderson's work, we could have expected comparable performance had our experimentally trained pilots spent even one-third as much time as traditionally trained pilots. In fact, all of the experimentally trained pilots completed the training in no more than five hours, which averaged approximately one-sixth as much time as traditionally trained pilots spent in training time – with no accommodation for individual differences in learning rate. (Some traditionally trained pilots spent as little as 10 hours on CDU practice while others spent as much as 50 hours, probably averaging about 30 hours.)

Not surprisingly, therefore, five hours turned out to be enough time to approximate but not fully match the performance level of the traditionally trained pilots. On the accuracy measure, the best index of performance, there was no consistent pattern of statistical significance favoring the traditionally trained group over the experimentally trained group, making the performance of the two groups technically equivalent. On the solution time measure the experimentally trained pilots performed consistently slower. Overall task times per keystroke were 55% longer for experimentally trained pilots than for the traditionally trained group, and there were significant between-group differences in both time per keystroke and time per task.

In our second experiment (see Section 4.5) the practice time for the “simulator” group using the actual flight hardware reduced the time-per-keystroke measures to levels that did were only 11% longer than traditionally trained pilots and not differ significantly from the time-per-keystroke measures of the traditionally trained pilot group. Nevertheless, the “simulator” pilots were younger than the experimentally trained pilots were, the sample size was small (n=6), and the time-per-keystroke differences between the “simulator” and traditionally trained groups approached statistical significance (p = .063). Therefore, it would be unjustifiable to conclude that 90 minutes of practice in a fixed-based simulator would ensure that experimentally trained pilots could perform the tasks as rapidly as the traditionally trained pilots, instead of taking significantly longer (see above, Section

4.2). It still appears necessary to also increase the total practice time for the experimental training program to make the performance of experimentally trained pilots equal to or greater than the performance of the traditionally trained.

We have concluded that five hours is not adequate training time. It would be better to extend the practice time until the per-keystroke times for our experimentally trained pilots were similar to the per-keystroke times of traditionally trained pilots. In reality, pilots are under considerable time pressure when performing inflight **Modify Route** tasks, and long per-keystroke times would surely compromise their performance on tasks that must be performed under time pressure. There is no doubt that time-per-keystroke would fall rapidly with additional practice trials. In skill acquisition experiments the solution/reaction time declines very steeply on the first few trials and then levels off, displaying a power function that exhibits a linear decline in solution time as a function of the log of the number of trials (Anderson, 1993, p. 53). In practical terms, the power-function relationship means that a few more practice trials per task would have a strong impact on lowering time-per-keystroke for the experimentally trained group.

Extending the number of practice trials on the harder tasks would also raise the percentage of experimentally trained pilots who could perform the difficult tasks without hints (Anderson, 1993). Most errors are due to missing pieces of knowledge (Anderson, 1993, p. 245), and other errors are due to acquiring buggy procedures. Experiments on skill-acquisition consistently show gains in novices' accuracy with increased practice, but accuracy does not necessarily improve with additional practice (Anderson & Schunn, 2000). Filling a learner's gaps in knowledge and/or correcting the learner's buggy procedures requires well-designed error feedback and practice solving particular problems that can only be solved by acquiring/debugging the condition-action rules that are missing/incorrect. For example, children who acquire buggy rules for math procedures often retain these buggy rules for a lifetime. The buggy rules produce the right answers just often enough that the child is never corrected for using the buggy rules, and each successive use makes the buggy rules stronger and more impervious to correction.

Performance criteria could be set for the experimentally trained group, requiring repetition of practice trials on harder tasks until the accuracy level and time-per-keystroke measure reached target levels. It is an open question what criteria should be set, how many additional trials per task would be required to meet these criteria, and whether meeting these criteria would be worth the expense in training time. Times per keystroke on the various CDU tasks would continue to fall towards an asymptote each time the pilot repeated performing CDU tasks during actual line experience after completion of training. Due to the year or more of line experience, the experts' mean times-per keystroke in Figure 9 are lower than the mean times-per keystroke of traditionally and experimentally trained pilots, showing how additional performance of the tasks increases the speed of performance. In Figure 8, however, experts' accuracy measures still showed evidence of gaps in knowledge or "buggy rules," suggesting that a year or more of additional line experience offered only a partial remedy for overcoming sources of difficulty and improving accuracy. Section 5.2 (below) describes more effective remedies for the sources of difficulty that marred the accuracy measures for all three groups of pilots.

Doubling experimental training time to 10 hours could double the number of practice trials on the harder tasks and make it possible to allocate more time to overcoming the sources of difficulty. A ten-hour version of experimental training would certainly have resulted in both higher accuracy and markedly higher performance speed, yet still be only approximately one-third of the estimated average time spent for traditional training on the CDU. The two-thirds savings in time (compared to traditional training) could then be allocated to covering new curriculum topics.

Because of dramatic gains in educational effectiveness, therefore, much more could be accomplished in the same amount of total training time that airline training programs now devote to teaching just 14 CDU tasks. This more effective training would teach all 14 CDU tasks currently covered in airline training programs plus a lot more topics: a far larger repertoire of CDU tasks and a better understanding of how the FMC "thinks" when carrying out the pilot's commands. Currently, essential topics that are not taught in the training program are covered during Initial Operating Experience (IOE). Line Check Airmen who administer the IOE all say that it would be better if the people coming to the line for IOE had a better understanding of how the FMC worked and how to use it. Airlines could reduce the on-the-job-training load by using this extra time to teach the pilots more about the FMC and how to use it in both L-NAV and V-NAV.

5.2 Design for enhanced performance on flightcrew-automation interface

The second major goal of our research was to design more effective aviation training based on the ACT-R skill acquisition theory and the eight principles for designing ACT-R intelligent tutors. These tutors have established a strong track record for yielding higher performance levels as well as reducing training time (Corbett, Koedinger, and Anderson, 1997; Anderson, Corbett, Koedinger, & Pelletier, 1995). Aiming for a higher performance level on the CDU interface would require an expanded curriculum, and Section 5.2.1 analyzes the value of using a fine-grained model of competence to define the curriculum (the first of the eight ACT-R principles). Section 5.2.2 discusses the selection of appropriate training equipment for the expanded curriculum. Section 5.2.3 shifts attention to an iterative process of training design that analyzes the performance of individuals who have completed the training program, using the resulting data to gradually refine both the fine-grained model of competence and the pedagogy designed to deliver the curriculum (the other seven ACT-R principles).

5.2.1 The value of representing CDU competence with a fine-grained model

Representation of competence with the NGOMSL model provided insight into the cognitive demands of the FMC in the limited context of the CDU tasks mandated for the FAA checkride. First, all tasks are carried out by means of a very limited number of common goals (**ACCESS, DESIGNATE,** and **INSERT**), and we emphasized these common goals in the way we initially taught the CDU tasks during the first session of the experimental training. The immediate error feedback also emphasized these common goals, but methods for achieving the common goals are heterogeneous and sometimes complex, creating a family of similar methods for accomplishing each goal. Capitalizing on these three common goals should make it simpler and more efficient to expand the repertoire of CDU tasks taught. The expanded repertoire should include, for example, vertical navigation tasks (V-NAV) and how to handle more complex clearances (e.g., crossing restrictions).

Second, all the inflight tasks mandated for the FAA checkride are highly similar and can be taught effectively by nesting them under the higher-level goal **Modify Route**. This calls pilots' attention to both the similarities and differences. Noticing the similarities fosters transfer. Highlighting the differences focuses pilots' attention on learning the selection rules that distinguish which particular **Modify Route** method to apply to comply with each particular ATC directive.

Third, we discovered inconsistencies in the CDU interface. The majority of the inconsistencies were anticipated by the initial NGOMSL analysis and later confirmed by comparing the NGOMSL model with the errors that pilots made performing the transfer test tasks in the full-motion simulator. Other inconsistencies were first revealed by pilot errors and subsequently confirmed by interpreting the errors in relation to the NGOMSL model.

In many cases, the inconsistency was due to a mismatch between the task defined by an Air Traffic Control clearance and the organization of the operations required to program the FMC to quickly carry out these directives. These task-mapping errors deserve serious additional study. Palmer et al. (1993) conclude that flightcrews encounter serious difficulties mapping ATC clearances to program the FMC. In all cases where an interface inconsistency problem can be solved by changes in the interface, improving the design of the CDU interface is the preferred path. Improving the consistency of the interface reduces pilot errors during flight and minimizes training time, because pilots do not have to learn to compensate for usability flaws in the interface.

In cases where the interface cannot be redesigned,[1] however, the experimental training should explicitly teach task mapping from ATC directives to the way the information must be programmed into the CDU interface in order to get the desired results from the FMC. For example, the pilot must re-map the clearance "INTERCEPT V83" to the CDU-compatible goal of "INTERCEPT the next down-path waypoint on V83." A second example is understanding the need to convert the radial (given in the ATC clearance) to its reciprocal in the INSTALL HOLD PARAMS task. Indeed, the experimental training for INSTALL HOLD PARAMS was an encouraging success case precisely because the experimental training so strongly emphasized the ATC-CDU mismatch and the method for translating the ATC clearance into the form the CDU would accept.

5.2.2 Training equipment: Role of current-generation desktop/laptop simulators

As discussed in Section 4.2, training deficiencies were partly responsible for the poor performance of the experimentally trained pilots on the INTERCEPT LEG-TO V83 and INTERCEPT FINAL APPROACH COURSE tasks. The desktop simulator used in the experimental training provided only a static representation of the HSI, and this resulted in knowledge gaps that lowered performance during the transfer test. In contrast, current-generation desktop/laptop simulators have the potential to prevent knowledge gaps. Using these modern simulators would facilitate acquisition of the full set of condition-action rules needed to use the HSI moving map display and MCP in order to perform CDU tasks, including but not limited to INTERCEPT LEG-TO tasks.

These desktop/laptop simulators have been touted recently for use as “free-play” tutors (e.g., Sherman and Helmreich, 1998). Generally these tutors can, however, also be used with lesson plans. Some modern simulators are designed to give instructors the option of creating various types of lesson plans to run the simulators – including lock-step, single-path CBT training – and several forms of error feedback (e.g., Aerosim, 2001b). Extrapolating from Anderson's research discussed earlier, the crucial problem with "free-play" learning comes from allowing the pilot to spend a high percentage of time floundering. In comparison, using the desktop/laptop simulators with a carefully designed syllabus would consume far less time and more reliably result in pilots mastering the full set of condition-action rules needed for competence (see Section 2.2.6 for elaboration and defense of this claim).

A logical extension of this research project would be to integrate modern desktop/laptop simulators with training design based on ACT-R skill acquisition theory and the eight ACT-R principles for designing intelligent tutors. The updated version of our experimental training would approximate a progression of training devices ranging from single-path part-task CBTs to more flexible part-task CBTs to high fidelity desktop/laptop simulations to full-motion simulators. This progression is validated by several of the ACT-R principles for training design, particularly the principles of gradually increasing the grain size of the instruction, providing immediate error feedback, and reducing the scaffolding to progress to real-world performance.

This extension of our research program would be unique in the way it nests cognitive task analysis within the ACT-R theory of skill acquisition and incorporates state-of-the-art desktop/laptop simulators as training devices. Aviation training researchers have independently arrived at many of the same conclusions as ACT-R researchers, but the research literature on training has developed in isolation from the research literature on modern theories of skill acquisition and transfer (see review by Salas & Cannon-Bowers, 2001). This is also true, more specifically, for the literature on training using simulators (Bell & Waag, 1998; Dennis & Harris, 1998; Fowlkes et al., 1998; Koonce & Bramble, 1998; Salas, Bowers, & Rhodenizer, 1998). This isolation has been unfortunate, because ACT-R has much to contribute to the theoretical foundations for research on

[1] For all the aircraft currently in use, the cost of redesigning and certifying CDU boxes is prohibitively expensive. It costs the airlines less to train pilots to work around the inconsistencies.

aviation training design and evaluating the effects of training on performance, as well as promising dramatic gains in the educational effectiveness of aviation training programs.

5.2.3 Iterative design to refine both the model that defines the curriculum and the pedagogy

Errors and confusion among students trained on early iterations of any ACT-R intelligent tutor have revealed condition-action rules missing from the first-pass model of competence, including failure to include all possible legitimate methods individuals may use to solve particular types of problems. Serious curriculum development requires filling in these missing rules (Anderson, 1993, p. 240). For example, analyzing data from student work on the geometry tutor revealed that stronger students, but not weaker students, developed a strategic plan for their geometry proof before constructing the proof. This insight spawned a decision to refine the model to include the strategic planning stage (Koedinger & Anderson, 1990, 1993). The process moves from theory to practice and back to theory, and each consecutive iteration of the ACT-R model and tutor has resulted in enhanced student performance.

In addition to building a more complete model of competence to define the curriculum for training design, consecutive iterations of ACT-R intelligent tutors have accomplished refinements in the pedagogy for delivering the curriculum, notably the design of error feedback (Anderson, Corbett, Koedinger, & Pelletier, 1995). The aviation community has emphasized task analysis, and task analysis is necessary but not sufficient to build a fine-grained model of competence. It is equally crucial to construct a complete model and to look at how well that model is transmitted to learners through the design of error feedback, amount of practice, and communication of the goal structure of the task.

For example, increasing the number of practice trials would inhibit negative transfer effects caused by inconsistencies in the interface or by frequency biases. Skill acquisition experiments demonstrate clearly that additional practice trials increase the strength of the condition-action rules required to perform the task, making skilled performance more accurate, not just faster (Anderson, 1993). For an example of reducing errors caused by interface inconsistencies, more explicit instruction on exceptions to the rule ought to correct the deficiencies in accuracy on the REVISE V-NAV CLIMB task. For an example of countering the frequency biases that cause errors, pilots might be taught to inhibit the frequency bias for performing **Modify Route** tasks on the LEGS page. The frequency bias develops because about 90% of all **Modify Route** tasks must be performed on the LEGS page. Since installing an airway is a less common **Modify Route** task that must be performed on the Route page, not the LEGS page, it requires inhibiting the frequency bias. If diverse instances of the INSTALL AIRWAY task were practiced disproportionately often during training it could help pilots instantly remember to counter the frequency bias and **ACCESS** the Route page when the ATC directive calls for installing an airway.

Of particular concern in our research are tasks on which the mean accuracy of experts fell below 90% (see Figure 8). It is crucial to recognize that even expert pilots – who had a year or more of line experience flying glass-cockpit aircraft – continued to manifest performance deficiencies on most of the same tasks that posed sources of difficulty for the experimentally and traditionally trained groups. These expert performance deficiencies demonstrate that merely increasing the number of trials per task during training would not adequately solve the problems created by sources of difficulty. Because such errors are unacceptable among expert line pilots, it is necessary to prioritize overcoming these expert difficulties when designing the next iteration of the experimental training.

Improving instruction on tasks with known sources of difficulty should begin by further decomposing the specific steps associated with the sources of difficulty – assuming that the model of competence is not complete. Some – albeit not all – experts are able to flawlessly perform these difficult steps, and it is important to start with a cognitive task analysis of exactly how they do it. Once understood, we need to ensure that the model of competent performance represents all the components required to best perform the task. The model of the task INSTALL HOLD PARAMS proved to be a success case, because it correctly represented the task mapping required to translate the information in the ATC directive into the form required by the FMC and successfully taught it to pilots. The goal of the second iteration of experimental training would be to turn the remaining difficult CDU tasks into comparable success cases.

Interactions between the FMC, its displays, and other aspects of automation in the "glass cockpit" were purposely omitted from our training design. Results of other investigators, however, make it very clear that these interactions are a major source of difficulty for pilots transitioning to "glass cockpit"/FMC aircraft, extending well beyond the time of exiting the training program into line operation. In addition, our own results show that experimentally trained pilots' performance fell because we failed to adequately teach pilots to use the HSI while performing the INTERCEPT tasks on the CDU. This is a special case of the ACT-R intelligent tutor design guideline that specifies constructing a complete fine-grained model of competence to define the curriculum.

As ACT-R research on learners' errors has shown (Anderson, 1993, p. 243), almost all errors are due to lack of knowledge, not to misconceptions or buggy procedures, so it is crucial to define the curriculum using a fully complete model of CDU task competence. We recommend developing a complete NGOMSL model of flightcrew competence for the total repertoire of CDU tasks that pilots must perform during flight, not just the 14 CDU tasks mandated for the FAA checkride, and to use this complete model to define the curriculum for future iterations of the CDU training. Accordingly, we recommend expanding the NGOMSL model to encompass use of the complete autoflight environment, including mode control panel (MCP), flight mode annunciations (FMAs), horizontal situation indicator (HSI), and pilots' comprehension of vertical as well as lateral navigation. In the important effort to address problems with vertical navigation (use of the FMC mode V-NAV), the complete model should include the existing displays as sources of information, and to promote robust performance of CDU tasks at all levels of automation – full, intermediate, and manual control.

When moving to the expanded curriculum, however, care must be taken to use a hierarchy of training environments that move from simple to complex. Curriculum and cognitive objectives should drive the selection of training devices. We followed the pedagogical principles of learning by solving problems, clearly communicating the goal structure, and supplying immediate feedback. We selected and sequenced problems in order to promote gradual acquisition and sufficient practice of each of the individual skills required for competence. With an expanded curriculum, it becomes both more challenging and more important to introduce the skills gradually, and to limit simulated displays to the hardware devices that are necessary for performing the skills being learned at the moment.

Greater realism in sophisticated displays of the HSI, MCP, map, etc., can actually distract learners' attention from acquiring the crucial skills represented in the fine-grained model of competence, failing, as a result, to provide the gains in learning that designers expected from such increased fidelity during training (Salas, Bowers, & Rhodenizer, 1998). Flightcrew training must initially teach tasks in a simple part-task training environment and culminate with training in a full-motion simulator in order to integrate performance of separate tasks learned and practiced in simple training environments.

In the training program described herein, we followed the pedagogical principles of gradually increasing the grain size of instruction, providing immediate feedback, and progressing to real-world performance (see above, Sections 2.2.4-2.2.6). We supported the progression in these guidelines by selecting a parallel progression of the specific training environment and training equipment. Training started with the single-path CBT part-task trainer in Session I, moved to the SuperCard™ desktop simulator with a carefully designed syllabus of problems nested in realistic flight scenarios, and culminated in the full-motion simulator transfer test. These design principles take on even greater importance for the more complete model of competence and correspondingly more complex curriculum outlined above.

1.3 Practical implications of this research

Aircraft equipped with advanced levels of automation have compiled safety records generally better than the safety records for less automated aircraft (Funk, 1997). Nevertheless, problems with the flightcrew-automation interface have contributed in varying degrees to many serious incidents and fatal crashes and aroused aircrew concerns about potential hazards (Billings, 1997; Eldredge, Mangold, & Dodd, 1992; Johnson & Pritchett, 1995; Mellor, 1994; Mosier, Skitka, & Korte, 1994; Palmer, Hutchins, Ritter, & VanCleemput, 1993; Sarter & Woods, 1995a, 1995b). As a result of the mounting evidence of credible but correctable threats to safety, the Federal Aviation Administration (FAA) chartered a Human Factors Team (FAA, 1996) to investigate the reported problems with interfaces between flightcrews and highly automated flightdeck systems and to make recommendations.

More evidence has appeared since publication of the 1996 FAA report (for example, Billings, 1997; Pasuraman & Riley, 1997; Sherman, 1997). Pilots continue to report difficulties with learning and carrying out tasks on aircraft employing advanced levels of automation (Air Transport Association, 1997, 1998, 1999; BASI, 1998). In addition, Funk and Lyall, and their colleagues are nearing completion of a major FAA-funded study to synthesize all the available evidence on problems with the flightcrew-automation interface (Funk, Lyall, and Suroteguh, 1999; Lyall, Niemczyk, & Lyall, in preparation; Owen & Funk, 1997; Wilson & Funk, 1997a, 1997b).

The goal of this research project described in this paper was to provide proof of concept for the possibility of designing a training program that could successfully teach the FAA-mandated CDU tasks in less time on less expensive equipment. Having accomplished this goal makes it feasible to redirect the savings in training time and training expense to expand the number of CDU tasks taught during training and to improve pilot understanding of the FMC and the CDU interface. Although we discovered several flaws in the first iteration of the training, all of these flaws could be corrected in subsequent iterations, still keeping equipment costs much lower and training time at about one-third the length of current (traditional) training. Therefore, it appears highly feasible to expand the repertoire of CDU tasks taught during the CDU training and also achieve a higher level of pilot performance on these tasks without exceeding the time and money that commercial airline companies currently invest in traditional training. Many aviation researchers, FAA officials, experienced pilots, and airline company representatives share the goals of (1) improving pilot understanding of the FMC, and (2) expanding the repertoire of CDU tasks taught during training (see, e.g., the extensive evidence compiled in the meta-analyses by Funk, Lyall, & Suroteguh, 1999). Fortunately, accomplishing these two goals appears to be well within our grasp with subsequent iterations of the experimental training tested in the FAA-funded research project we have described in this report.

APPENDIX: ONE EXAMPLE OF THE TRAINING SCENARIOS

1076 ORD-RSW

ORD..EON..DNV.J73.TLH.J41.RSW..RSW

NAME	ID	CRS	DIST	SPD/ALT
CHICAGO	ORD			
PEOTONE	EON			
DANVILLE	DNV			.75/33000
JEANE		161	28	.75/33000
MCVIK			22	.75/33000
TERRE HAUTE	TTH		8	.75/33000
DUSTS		170	163	.75/33000
LONGE			6	.75/33000
NASHVILLE	BNA	174	35	.75/33000
LA GRANGE	LGC	159	198	.75/37000
GOONS		163	74	.75/37000
TALLAHASSEE	TLH	162	81	.75/37000
LICKS		148	51	.75/37000
LEGGT			41	.75/37000
TABIR			55	.75/37000
ST PETERSBURG	PIE	150	35	
LEE COUNTY	RSW	150	96	
FT MYERS	RSW			

1. After takeoff on 32L at ORD the flight has been given a turn to a heading of one-six-zero. The flight is 20 miles southwest of Chicago when ATC says, "United 1076 cleared direct Peotone (EON), flight plan route."
2. Between Peotone (EON) and Danville (DNV), ATC issues the clearance, "United 1076 cleared direct Indianapolis (VHP), direct Louisville (IIU), direct Nashville (BNA), flight plan route."
3. Approaching Nashville, ATC says, "United 1076, after Nashville cleared direct St Petersburg (PIE)."

   At St Petersburg, you get the following clearance: "United 1076, turn right to a heading of one-eight-zero and intercept the 273 degree radial to La Belle (LBV)."
4. Now ATC says, "United 1076 hold northwest of La Belle (LBV) on the 273˚ radial, left turns. Expect further clearance at 1912 Zulu. Time now 1844."

REFERENCES

Aerosim Technologies (2001a). *Products.* Available at http://www.aerosim.com/products.htm

Aerosim Technologies (2001b). *Lesson plan editor*. Available at http://www.aerosim.com/lpe.htm.

Air Transport Association (1997). *Towards an operational philosophy and model training program for FMC-generation aircraft* (First report by the Human Factors Committee Automation Subcommittee). Copies of this report will be provided electronically upon e-mail request to Tom_Chidester@amrcorp.com.

Air Transport Association (1998*). Potential knowledge, policy or training gaps regarding operation of FMS-generation aircraft* (Second report by the Human Factors Committee Automation Subcommittee). Copies of this report will be provided electronically upon e-mail request to Tom_Chidester@amrcorp.com.

Air Transport Association (1999). *Performance of standard navigation tasks by FMS-generation aircraft* (Third report by the Human Factors Committee Automation Subcommittee). Copies of this report will be provided electronically upon e-mail request to Tom_Chidester@amrcorp.com.

Anderson, J. R. (1993). *Rules of the mind.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Anderson, J. R., Corbett, A. T., Koedinger, K. R., & Pelletier, R. (1995). Cognitive tutors: Lessons learned. The *Journal of the Learning Sciences, 4*(2), 167-207.

Anderson, J. R., & Lebiere, C. (1998). *The atomic components of thought*. Mahwah, NJ: Lawrence Erlbaum Associates.

Anderson, J. R., & Schunn, C. D. (2000). Implications of the ACT-R learning theory: No magic bullets. In R. Glaser (Ed.), *Advances in instructional psychology: Vol. 5. Educational design and cognitive science* (pp. 1-33). Mahwah, NJ: Lawrence Erlbaum Associates.

BASI (1998) *Advanced Technology Aircraft Safety Survey Report* (Report by Department of Transport and Regional Development, Bureau of Air Safety Investigation). Available from the website for the Australian Transport Safety Bureau (ATSB) Bureau of Air Safety Investigation (BASI) at http://www.basi.gov.au/sprog/advtek.htm.

Bell, H. H., & Waag, W. L. (1998). Evaluating the effectiveness of flight simulators for training combat skills: A review. *International Journal of Aviation Psychology, 8*(3), 223-242.

Billings, C. E. (1997). *Aviation automation: The search for a human-centered approach.* Mahwah, NJ: Lawrence Erlbaum Associates.

Chi, M. T. H., Bassok, M., Lewis, M. W., Reimann, P., & Glaser, R. (1989). Self-explanations: How students study and use examples in learning to solve problems. *Cognitive Science, 13,* 145-182.

Chi, M. T. H., de Leeuw, N., Chiu, M-H., & LaVancher, C. (1994). Eliciting self-explanations improves understanding. *Cognitive Science, 18,* 439-477.

Corbett, A. T., Koedinger, K. R., & Anderson, J. R. (1997). Intelligent tutoring systems. In M. Helander, T. K. Landauer, & P. Prabhu (Eds.), *Handbook of human-computer interaction* (2nd, completely revised edition, pp. 849-874). Amsterdam: Elsevier.

Dennis, K. A., & Harris, D. (1998). Computer-based simulation as an adjunct to ab initio flight training. *International Journal of Aviation Psychology, 8*(3), 261-276.

Eldredge, D., Mangold, S., & Dodd, R. S. (1992). *A Review and Discussion of Flight Management System Incidents Reported to the Aviation Safety Reporting System.* **(**Final Report -

DOT/FAA/RD-92/2.) Washington: U.S. Department of Transportation - Federal Aviation Administration.

Eurocontrol (2000). *Guidelines for the Production of Computer Based Training.* Available on the Eurocontrol website at http://www.eurocontrol.be/projects/ians/cbtguidelines/html/body_index.html.

Federal Aviation Administration (FAA) Human Factors Team (1996*). Report on the interfaces between flightcrews and modern flight deck systems* (June 18, 1996). Washington: U.S. Department of Transportation, Federal Aviation Administration.

Fowlkes, J., Dwyer, D. J., Oser, R. L., & Salas, E. (1998). Event-based approach to training (EBAT). *International Journal of Aviation Psychology, 8*(3), 209-221.

Funk, K. (1997). *Aircraft accident rates: Conventional vs. advanced technology.* Available on the Flight Deck Automation Issues website at http://flightdeck.ie.orst.edu/FDAI/hullLossRates.html.

Funk, K., Lyall, E., & Suroteguh, C. (1999). *Meta-analysis.* Available on the Flight Deck Automation Issues website at http://flightdeck.ie.orst.edu/FDAI/Meta-Analysis/meta-analysis.html.

Gopher, D., Weil, M., & Bareket, T. (1994). Transfer of skill from a computer game trainer to flight. *Human Factors, 36*(3), 387-405.

Jentsch, F., & Bowers, C. A., (1998). Evidence for the validity of PC-based simulations in studying aircrew coordination. *International Journal of Aviation Psychology, 8*(3), 243-260.

John, B. E., & Kieras, D. E. (1996a). Using GOMS for user interface design and evaluation: Which technique? *ACM Transactions on Computer-Human Interaction, 3*(4), 287-319.

John, B. E., & Kieras, D. E. (1996b). The GOMS family of user interface analysis techniques: Comparison and contrast. *ACM Transactions on Computer-Human Interaction, 3*(4), 320-351.

Johnson, E. N., & Pritchett, A. (1995). *Experimental study of vertical flight path mode awareness.* Cambridge, MA: Department of Aeronautics and Astronautics, MIT.

Kieras, D. (1997). A guide to GOMS model usability evaluation using NGOMSL. In M. G. Helander, T. K. Landauer, & P. V. Prabhu (Eds.), *Handbook of human-computer interaction* (2nd rev. ed., pp. 733-766). Amsterdam: Elsevier.

Koedinger, K. R., & Anderson, J. R. (1990). The role of abstract planning in geometry expertise. *Cognitive Science, 14,* 511-550.

Koedinger, K. R., & Anderson, J. R. (1993). Reifying implicit planning in geometry: Guidelines for model-based intelligent tutoring system design. In S. P. Lajoie & S. J. Derry (Eds.), *Computers as cognitive tools.* Hillsdale, NJ: Lawrence Erlbaum Associates.

Koonce, J. M., & Bramble, W. J., Jr. (1998). Personal computer-based flight training devices. *International Journal of Aviation Psychology, 8*(3), 277-292.

Longridge, T. M. (2000). *Overview of the Advanced Qualification Program.* Available on the Federal Aviation Administration website at http://www.faa.gov/avr/afs/tlpaper.htm.

Lyall, E., Niemczyk, M., & Lyall, R. (in preparation, tentative title). *Evidence for Flightdeck Automation Problems: a Survey of Experts.* Summary of evidence from the study is available on the Flight Deck Automation Issues website at http://flightdeck.ie.orst.edu/scripts/dbsqldev/evidstudy.idc?study=93.

Mellor, P. (1994). *CAD: Computer aided disaster!* Unpublished manuscript. London: City University, Centre for Software Reliability.

Mosier, K.L., Skitka, L.J., & Korte, K.J. (1994). Cognitive and Social Psychological Issues in Flight Crew/Automation Interaction. *Proceedings of the First Automation Technology and Human*

Performance Conference (Washington, DC April 7-9, 1994). Hillsdale, NJ: Lawrence Erlbaum Associates, Publishers.

Newell, A. (1980). Reasoning, problem solving, and decision processes: The problem space as a fundamental category. *Attention and Performance, 8,* 693-718.

Owen, G., & Funk, K. (1997). *Incident Report Analysis.* Available on the Flight Deck Automation Issues website at http://flightdeck.ie.orst.edu/FDAI/IncidentStudy/incident.html.

Palmer, E. A., Hutchins, E. L., Ritter, R. D., & VanCleemput, I. (1993). *Altitude Deviations: Breakdown of an Error-Tolerant System* (NASA Technical Memorandum 108788). Field, CA: NASA Ames Research Center.

Parasuraman R., & Riley, V (1997). Humans and Automation: Use, misuse, disuse, abuse. *Human Factors, 39* (2), 230-253.

Pennington, N., Nicolich, R., and Rahm, J. (1995). Transfer of training between cognitive subskills: Is knowledge use specific? *Cognitive Psychology, 28,* 175-224.

Polson, P. G, Irving, S., and Irving, J. E. (1995). *Final Report: Applications of Formal Models of Human-Computer Interaction to Training and Use of the Control and Display Unit.* (Federal Aviation Administration, Contract No. DTFA01-91-00028.)

Salas, E., & Cannon-Bowers, J. A. (2001). The science of training: A decade of progress. *Annual Review of Psychology, 52,* 471-499.

Salas, E., Bowers, C. A., & Rhodenizer, L. (1998). It is not how much you have but how you use it: Toward a rational use of simulation to support aviation training. *International Journal of Aviation Psychology, 8*(3), 197-208.

Sarter, N. B., & Woods, D. D. (1995a). *Strong, Silent, and Out-of-the-loop: Properties of Advanced (Cockpit) Automation and Their Impact on Human-Automation Interaction.* (CSEL report 95-TR-01.).

Sarter, N. B., Woods, D. D. (1995b). 'How in the world did we ever get into that mode?' Mode error awareness in supervisory control. *Human Factors 37*(1), 5-19.

Schraagen, J. M., Chipman, S. F., & Shalin, V. L. (Eds.) (2000). *Cognitive task analysis.* Mahwah, NJ: Lawrence Erlbaum Associates.

Seamster, T. L., Redding, R. E., & Kaempf, G. L. (1997). *Applied cognitive task analysis in aviation.* Aldershot, UK and Brookfield, VT: Avebury Aviation.

Sherman, P. J. (1997). *Aircrews' evaluations of flight deck automation training and use: Measuring and ameliorating threats to safety.* (The University of Texas Aerospace Crew Research Project, Technical Report 97-2).

Sherman, P. J., & Helmreich, R. L. (1998). Training for use of automation: The value of "free-play" as a complement to traditional transition training. In *Proceedings of the Ninth International Symposium on Aviation Psychology* (pp. 243-2 48). Columbus, OH: The Ohio State University. Also available at http://www.psy.utexas.edu/psy/helmreich/osu97pjs.htm.

Tricom Technologies (2001). *Products.* Available on the Tricom Technologies website at http://www.tricom-tech.com/products.htm.

VanLehn, K. A., Jones, R.M., & Chi, M.T.H. (1992). A model of the self-explanation effect. *Journal of the Learning Sciences, 2,* 1-59.

Wicat (2001a). *Simulations and training devices.* Available on the Wicat website at http://www.wicat.com/flight/simstrns/simstrns.htm.

Wicat (2001b). *Simulations and trainers for flight and maintenance training.* Available on the Wicat website at http://www.wicat.com/wicat/sims.htm.

Wilson, J., & Funk, K. (1997a). *Accident analysis.* Available on the Flight Deck Automation Issues website at http://flightdeck.ie.orst.edu/FDAI/AccidentStudy/accident.html.

Wilson, J., & Funk, K. (1997b). *Experiments, Surveys, and Other Studies Analysis.* Available on the Flight Deck Automation Issues website at http://flightdeck.ie.orst.edu/FDAI/StudyStudy/study.html.